Text To Speech Based on Transformer

Written by Guowen Wang 36920221153118, Wujin Sun 36920221153114, Qingyao Wu 36920221153124, Rigong Te 36920221153116, Mingliang Huang 36920221153083 (AI class)

Electronic Information Major, Artificial Intelligence Research Institute, Xiamen University

Abstract

We found that the mel-spectrogram generated by FastSpeech[15] has shadow and missing mid-to-high frequencies, which will result in a sound that is too smooth without emotional ups and downs. We studied the structure of this model and found that although the Transformer structure adopted by FastSpeech improves the quality of synthesized speech and speeds up the synthesis speed, but the Transformer structure pays too much attention to the global information. So a certain part of the local information is ignored and the synthesized sound is too flat. To solve this problem, we introduce parallel residual block and convolution block to extract local information. At the same time we add two linear layers at the end of the decoder to solve the problem of energy and pitch lossing. This structure is only used in the training phase. The experiments on the LJSpeech dataset shows that our model improves the quality of the sound and speeds up sound synthesis.

Information

Text to speech is a technology that converts text into speech. Speech synthesis plays an indispensable role in people's lives and plays an important role in human-computer interaction. With the development of deep learning, speech synthesis technology based on deep learning surpasses traditional speech synthesis technology in both the quality and speed of synthesized speech. Some studies have introduced transformers into experiments to verify that they can generate more realistic sounds than traditional methods.But there is still the problem of skipping words and repeating words in particularly difficult cases, so FastSpeech[15] extracts the attention alignment in the encoder-decoder model, using a length regulator to expand the source phoneme sequence to match the target mel-spectrogram sequence to generate parallel mel-spectrogram,thus solved the problem.

However, the mel-spectrogram generated by FastSpeech has the problem of artifacts, because of the lack of high-frequency information, and the energy and pitch information will be lost during the decoding process. The reason for this problem is that the attention mechanism focuses on global information, which makes the resulting image is over-smoothed, and the language becomes stiff, undulating, and lacking in realism. Therefore, we introduced a parallel residual block and convolution block. Through this method, the network will pay attention to the parts that it values while paying attention to the overall situation. For the problem of energy and pitch loss, we use a linear layer at the end of the decoder to deal with it, so that the network can predict the decoded energy and pitch. This is only used in the training phase, and this part is removed in the testing phase.

We use the LJSpeech dataset to evaluate our model and previous models and find that our model is faster and more realistic than FastSpeech. In addition, on this basis, we further use a non-autoregressive text-to-waveform generation model, which has the advantage of complete end-to-end inference and achieves faster inference speed. Our experimental results show that our model approach is improved, especially in terms of speech quality. The training pipeline is much simpler, and the speech synthesis speed of the model is fast.

Related work

Text to speech (TTS), also known as speech synthesis, aims to synthesize accurate natural speech given text[18], and is a hot research topic in the fields of natural language and deep learning. With the rapid development in the fields of artificial intelligence, natural language and speech processing, the combination of deep neural networks and TTS has brought about a significant improvement in the quality of speech synthesis. In recent years, with the popularization of information equipment and the advent of the Internet era, voice services such as voice calls, voice assistants, short video dubbing and other functions can enrich and facilitate people's lives, which is of great significance to promoting social development.

In the 2nd half of the 18th century, people wanted to create machines to synthesize human speech. In the second half of the 18th century, the Hungarian scientist Wolfgang von Kempelen built a speaking machine that could generate simple words and short sentences. In the second half of the 20th century, the first computer-based speech synthesis systems were introduced.Early computer-based speech synthesis methods include articulatory synthesis[17], formant synthesis and concatenative synthesis. Later, with the development of statistical machine learning, Statistical Parametric Speech Synthesis (SPSS) [19]was proposed, and

speech synthesis was further developed. Statistical parametric speech synthesis methods are able to predict speech synthesis parameters such as spectrum, fundamental frequency and duration. Since the 2010s, neural network-based speech synthesis[10][21] has achieved better speech quality and has gradually become the mainstream method. Some works introduce deep neural network into SPSS, such as those based on deep neural network (DNN)[13] and those based on recurrent neural network (RNN)[25]. However, these models replace the HMM with a neural network and still predict acoustic features from linguistic features, following the paradigm of SPSS. Later, Wang et al.[20] proposed to generate acoustic features directly from phoneme sequences instead of linguistic features, which can be regarded as the first exploration of end-to-end speech synthesis.

The neural network-based speech synthesis has high speech quality in terms of intelligibility and naturalness, outperforms traditional cascade and statistical parameter methods, and requires less human preprocessing and feature development. However, training TTS models in an end-to-end manner suffers from the different modalities between text and speech waveforms, as well as the huge length mismatch between character/phoneme sequences and waveform sequences. There are many challenges:

- **Slow speech synthesis inference speed.** Because mel-spectrogram sequences are typically hundreds or thousands in length, autoregressive models are slower to reason when decoding to generate mel-spectrograms.
- **Synthesized speech lacks robustness and controllability.** Due to error propagation[1] and wrong attention alignment between text and speech in autoregressive generation, the generated mel-spectrogram usually suffers from word skipping and repetition problems[2]. Meanwhile, it is often difficult to directly control speech rate and prosody in autoregressive generation.
- **The training of short audio clips corresponding to partial text sequences impairs text feature extraction.** Due to the limitation of waveform sample length and GPU memory, we can only train short audio clips corresponding to partial text sequences, which makes it difficult for the model to capture the relationship between phonemes in different partial text sequences.

Text to Speech TTS, which aims to synthesize natural and understandable speech of a given text, has been a hot research topic in the field of artificial intelligence for a long time. The research of TTS has shifted from the early stage of stage synthesis and statistical parameter synthesis to parameter synthesis based on neural networks and end-to-end models, and the speech quality of end-to-end model synthesis is close to the parity of human beings. The end-to-end TTS model based on neural network usually first converts text into acoustic features (such as mel-spectrograms), and then converts mel-spectrograms into audio samples. However, most neural TTS systems generate mel-spectrograms in an autoregressive manner, which is slow in reasoning and lacks robustness and controllability of synthetic speech.

Text Analysis It is used for the extraction of text conversion feature information for subsequent speech synthesis. In statistical parametric synthesis, text analysis includes several functions such as text normalization[26], word segmentation[22], part-of-speech (POS) tagging[16], prosody prediction[3], and phoneme conversion[24]. Neural TTS greatly simplifies the text analysis module at the stage of text analysis, modeling the input characters and phonemes. Faced with any possible non-standard raw samples, Neural TTS also requires text normalization to obtain standard word formats from character input, and also requires grapheme-to-phoneme conversion to obtain phonemes from standard word formats, more suitable for real-life use.

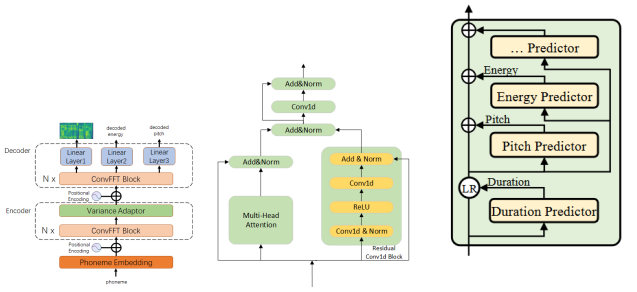
Acoustic model The role of the acoustic model is to directly generate acoustic features from language features, phonemes or characters, which are further transformed into waveforms using a vocoder. TTS acoustic models include early HMM-based and DNN-based models[25][4] in Statistical Parametric Speech Synthesis (SPSS), followed by sequence-to-sequence models[8]based on an encoder-attention-decoder framework, and state-of-the-art feedforward networks[11][15] for parallel generation. The choice of acoustic features largely determines the type of TTS pipeline. Tried acoustic features such as mel cepstral coefficient (MCC), mel generalized coefficient (MGC), band aperiodic (BAP), fundamental frequency (F0), voiced/unvoiced (V/UV), bark-frequency cepstral coefficients and the most widely used mel-spectrogram. The acoustic model is divided into two stages: 1) the acoustic model in SPSS, which usually predicts acoustic features based on linguistic features, such as MGC, BAP, and F0; 2) the acoustic model in neural-based end-to-end TTS, which is based on phonemes or Character predicted acoustic features (such as mel spectrograms).The attention mechanism is similar to the observation mechanism of human beings on external things, that is, people tend to focus on some important local information in a lot of information, and choose information that is more critical to the current thing to form an overall view of thing's impression. In recent years, attention mechanisms have been widely used in fields such as nlp and CV. The attention mechanism can be divided into soft attention and strong attention, in which soft attention pays more attention to channels or regions and is differentiable. In a neural network model, the attention mechanism usually takes the form of an additional neural network, which can help the model rigidly select certain parts of the input, or assign different weights to different parts of the input. The basic idea of the attention mechanism is to use the feature map to learn the weight distribution, and then apply the learned weight to the original feature map for weighted summation. Now attention machines are also used in TTS.

Vocoder The development of vocoders can be divided into two stages: vocoders for statistical parametric speech synthesis (SPSS), and vocoders based on neural networks. The vocoder includes steps of vocoder analysis and vocoder synthesis. In vocoder analysis, it analyzes speech and obtains acoustic features such as Mel cepstral coefficients, fre-

quency band aperiodicity, and F0. In vocoding, it generates speech waveforms based on these acoustic features. Neural vocoders fall into different categories: 1) autoregressive vocoders, 2) Flow-based vocoders, 3) GAN-based vocoders, 4) VAE-based vocoders and 5) diffusion-based vocoders . Early neural vocoders, such as WaveNet[9], Char2Wav, WaveRNN[5], directly take language features as input and generate waveforms. Later, Prenger et al.[12], Kim et al.[6], Kumar et al.[7], Yamamoto et al[23]. took Mel spectrograms as input and generated waveforms. Since speech waveforms are long, autoregressive waveform generation requires a lot of inference time. Therefore, generative models are used in waveform generation.

Proposed Solution

In order to improve the slow inference speed, lack of robustness and controllability of the current TTS model, we introduce the variance adaptor module.The variance adapter consists of a duration predictor, a pitch predictor and an energy predictor.



(a) Model. (b) ConvFFT Block. (c) Variance adaptor.

Figure 1: The overall architecture for our model.(a).The Convolution feed-forward Transformer.(b).The Variance adaptor.(c)

Motivation The mel-spectrogram generated by FastSpeech has the problem of artifacts, because of the lack of high-frequency information. The reason for this problem is that the attention mechanism focuses on global information, which makes The resulting image is over-smoothed, and the language becomes stiff, undulating, and lacking in realism. Therefore, we introduced a parallel residual block and convolution block. Through this method, the network will pay attention to the parts that it values while paying attention to the overall situation. For the problem of energy and pitch loss, we use a linear layer at the end of the decoder to deal with it, so that the network can predict the decoded energy and pitch. This is only used in the training phase, and this part is removed in the testing phase.

Model Overview The model is shown in Figure a.We notice that FastSpeech2[14] only compute energy loss and pitch loss in Variance Adaptor, while ignoring informations of energy and pitch might partially loss during the process of decoding. To solve the problem of inconsistancy of energy and pitch, we propose additional energy and pitch losses in decoder. As illustraed in figure 1(a) we add two extra linear layers in the end of decoder to predict decoded energy and pitch. But these two additional outputs is only used in training phase to improve the decoder's ability to maintain information.

ConvFFT Block A subtle artifact can be observed in mel-spectrogram generated by FastSpeech, which is that generated sample is detail lacked in high frequency parts. This problem is caused for only global informations is extracted by multi-head attention block,Since this over-smoothing problem can lead to unnaturalness in speech, we propose a residual convolution block in parallel with multi-head attention block to extract local information which aims to achive a trade-off between global and local inforormation. As shown in figure 1(b), same vector is input into multi-head attention block and residual conv1d block to extract global and local information respectively, then the two extracted information vectors are fused together. Feed-Forward Transformer stacks multiple FFT blocks for phoneme to mel-spectrogram transformation, with N blocks on the phoneme side, and N blocks on the mel-spectrogram side, with a length regulator (which will be described in the next subsection) in between to bridge the length gap between the phoneme and mel-spectrogram sequence. Each FFT block consists of a self-attention and 1D convolutional network, as shown in Figure 1b. The self-attention network consists of a multi-head attention to extract the cross-position information. Different from the 2-layer dense network in Transformer, we use a 2-layer 1D convolutional network with ReLU activation. The motivation is that the adjacent hidden states are more closely related in the character/phoneme and mel-spectrogram sequence in speech tasks. We evaluate the effectiveness of the 1D convolutional network in the experimental section. Following Transformer residual connections, layer normalization, and dropout are added after the self-attention network and 1D convolutional network respectively.

Variance adaptor In training, we take ground-truth values of duration, pitch, and energy extracted from the recording as input to the hidden sequence to predict the target voice. At the same time, we use ground true duration, pitch, and energy as targets to train duration, pitch, and energy predictors, which are used for inference to synthesize target speech. The variance adaptor aims to add variance information to the phoneme hidden sequence, which can provide enough information to predict variant speech for the one-to-many mapping problem in TTS. We briefly introduce the variance information as follows: 1) phoneme duration, which represents how long the speech voice sounds; 2) pitch, which is a key feature to convey emotions and greatly affects the speech prosody; 3) energy, which indicates framelevel magnitude of mel-spectrograms and directly affects the volume and prosody of speech. More variance information can be added in the variance adaptor, such as emotion, style and speaker, and we leave it for future work. Correspondingly, the variance adaptor consists of 1) a duration predictor 2) a pitch predictor, and 3) an energy predictor, as shown in

Figure c. In training, we take the ground-truth value of duration, pitch and energy extracted from the recordings as input into the hidden sequence to predict the target speech. At the same time, we use the ground-truth duration, pitch and energy as targets to train the duration, pitch and energy predictors, which are used in inference to synthesize target speech.

- **Duration Predictor** The duration predictor takes the phoneme hidden sequence as input and predicts the duration of each phoneme, which represents how many mel frames correspond to this phoneme, and is converted into logarithmic domain for ease of prediction. The duration predictor is optimized with mean square error (MSE) loss, taking the extracted duration as training target. Instead of extracting the phoneme duration using a pretrained autoregressive TTS model in FastSpeech, we use Montreal forced alignment (MFA) tool to extract the phoneme duration, in order to improve the alignment accuracy and thus reduce the information gap between the model input and output.
- **Pitch Predictor** Previous TTS systems with tone prediction based on neural networks usually directly predict tone profiles. In order to better predict the change of pitch contour, we use continuous wavelet transform to decompose the continuous tone sequence into a tone spectrogram, and take the tone spectrogram as the training target of the tone predictor, which uses MSE loss to optimize. In the inference, the pitch predictor predicts the pitch spectrogram and further converts it into the echo contour using the inverse continuous wavelet transform.
- **Energy Predictor** We calculate the L2 norm of the amplitude of each short-time Fourier transform frame as the energy. Then, we quantize the energy of each frame into 256 possible values, encode them as energy embedded e, and add them to the extended hidden sequence, similar to tones. We use the energy predictor to predict the original value of energy rather than the quantized value, and use MSE loss to optimize the energy predictor.

Experiments

Datasets We evaluate the model on LJSpeech dataset. LJSpeech contains 13,100 English audio clips (about 24 hours) and corresponding text transcripts. We split the dataset into three sets: 12,228 samples for training, 349 samples (with document title LJ003) for validation and 523 samples (with document title LJ001 and LJ002) for testing. For subjective evaluation, we randomly choose 100 samples in test set. To alleviate the mispronunciation problem, we convert the text sequence into the phoneme sequence with an open-source grapheme-to-phoneme tool5.

Model Configuration Our model addresses the issue of artifacts in the mel-spectrogram by introducing residual blocks and convolutional blocks. The energy and pitch loss issues are addressed by introducing a linear layer in the decoder.

Results To evaluate the perceptual quality, we perform mean opinion score (MOS) evaluation on the test set. Twenty

Model	Tradining Time(h)	Inference Speedup
Transformer TTS	38.64	/
FastSpeech	53.12	48.5×
Ours	**27.02**	**51.8**×

Table 1: The comparison of training time and inference latency in waveform synthesis. The training time of FastSpeech includes teacher and student training. The training and inference latency tests are conducted on a server with a NVIDIA 2080Ti GPU and batch size of 48 for training and 1 for inference.

Method	MOS
GT	4.30±0.07
GT (Mel + PWG)	3.92±0.08
Tacotron 2 (Mel + PWG)	3.70±0.08
Transformer TTS (Mel + PWG)	3.72±0.07
FastSpeech (Mel + PWG)	3.68±0.09
Ours	3.71±0.09

Table 2: Audio quality comparison.

native English speakers are asked to make quality judgments about the synthesized speech samples. The text content keeps consistent among different systems so that all testers only examine the audio quality without other interference factors. We compare the MOS of the audio samples generated by our model with other systems, including 1) GT, the ground-truth recordings; 2) GT (Mel + PWG), where we first convert the ground-truth audio into mel-spectrograms, and then convert the mel-spectrograms back to audio using Parallel WaveGAN(PWG); 3) Tacotron 2 (Mel + PWG); 4) Transformer TTS (Mel + PWG); 5) FastSpeech (Mel + PWG). All the systems in 3), 4) and 5) use Parallel WaveGAN as the vocoder for a fair comparison. The results are shown in Table 2. can match the voice quality of autoregressive models Transformer TTS and Tacotron 2. Importantly, our model outperforms FastSpeech, which demonstrates the effectiveness of providing variance information such as pitch, energy and more accurate duration and directly taking ground-truth speech as training target without using teacher-student distillation pipeline.

Conclusion

In this work, we address two issues in the model by adding parallel residual blocks and convolutional layers, and adding two linear layers in the decoder: 1) Artifacts exist in the mel-spectrogram, which leads to The voice is too stiff, without ups and downs. 2) Energy and pitch information loss during decoding.By introducing these two modules, the authenticity of the sound quality is improved, the fluctuation of the sound quality is increased, and the loss of energy and tone is alleviated.In addition, on this basis, we further use a non autoregressive text to waveform generation model, which has the advantage of complete end-to-end reasoning and achieves faster reasoning speed. Our experimental results show that our model method has been improved, es-

pecially in speech quality, it can even surpass the autoregressive model. The training pipeline is much simpler, and it inherits the advantages of the original TTS model of fast, robust and controllable speech synthesis. High quality, fast and completely end-to-end training without any external library is definitely the ultimate goal of neural TTS, and it is also a very challenging problem. In order to ensure the high quality of the model, we used external high-performance alignment tools and tone extraction tools. This seems a bit complicated, but it is very helpful for high-quality and fast speech synthesis. We believe that there will be a simpler solution to achieve this goal in the future, and we will certainly work on a complete end-to-end TTS without external alignment models and tools. We will also consider more variance information to further improve speech quality, and use lighter models to speed up reasoning.

References

[1] Ai, Y.; and Ling, Z.-H. 2020. A neural vocoder with hierarchical generation of amplitude and phase spectra for statistical parametric speech synthesis. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28: 839–851.

[2] Beliaev, S.; Rebryk, Y.; and Ginsburg, B. 2020. TalkNet: Fully-Convolutional Non-Autoregressive Speech Synthesis Model. *arXiv preprint arXiv:2005.05514*.

[3] Chu, M.; and Qian, Y. 2001. Locating boundaries for prosodic constituents in unrestricted Mandarin texts. In *International Journal of Computational Linguistics & Chinese Language Processing, Volume 6, Number 1, February 2001: Special Issue on Natural Language Processing Researches in MSRA*, 61–82.

[4] Fan, Y.; Qian, Y.; Xie, F.-L.; and Soong, F. K. 2014. TTS synthesis with bidirectional LSTM based recurrent neural networks. In *Fifteenth annual conference of the international speech communication association*.

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

[6] Kim, S.; Lee, S.-g.; Song, J.; Kim, J.; and Yoon, S. 2018. FloWaveNet: A generative flow for raw audio. *arXiv preprint arXiv:1811.02155*.

[7] Kumar, K.; Kumar, R.; de Boissiere, T.; Gestin, L.; Teoh, W. Z.; Sotelo, J.; de Brébisson, A.; Bengio, Y.; and Courville, A. C. 2019. Melgan: Generative adversarial networks for conditional waveform synthesis. *Advances in neural information processing systems*, 32.

[8] Li, N.; Liu, S.; Liu, Y.; Zhao, S.; and Liu, M. 2019. Neural speech synthesis with transformer network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 6706–6713.

[9] Oord, A.; Li, Y.; Babuschkin, I.; Simonyan, K.; Vinyals, O.; Kavukcuoglu, K.; Driessche, G.; Lockhart, E.; Cobo, L.; Stimberg, F.; et al. 2018. Parallel wavenet: Fast high-fidelity speech synthesis. In *International conference on machine learning*, 3918–3926. PMLR.

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

[11] Peng, K.; Ping, W.; Song, Z.; and Zhao, K. 2020. Non-autoregressive neural text-to-speech. In *International conference on machine learning*, 7586–7598. PMLR.

[12] Prenger, R.; Valle, R.; and Catanzaro, B. 2019. Waveglow: A flow-based generative network for speech synthesis. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3617–3621. IEEE.

[13] Qian, Y.; Fan, Y.; Hu, W.; and Soong, F. K. 2014. On the training aspects of deep neural network (DNN) for parametric TTS synthesis. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3829–3833. IEEE.

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

[15] Ren, Y.; Ruan, Y.; Tan, X.; Qin, T.; Zhao, S.; Zhao, Z.; and Liu, T.-Y. 2019. Fastspeech: Fast, robust and controllable text to speech. *Advances in Neural Information Processing Systems*, 32.

[16] Schlünz, G. I. 2010. *The effects of part–of–speech tagging on text–to–speech synthesis for resource–scarce languages*. Ph.D. thesis, North-West University.

[17] Shadle, C. H.; and Damper, R. I. 2001. Prospects for articulatory synthesis: A position paper. In *4th ISCA Tutorial and Research Workshop (ITRW) on Speech Synthesis*.

[18] Taylor, P. 2009. *Text-to-speech synthesis*. Cambridge university press.

[19] Tokuda, K.; Nankaku, Y.; Toda, T.; Zen, H.; Yamagishi, J.; and Oura, K. 2013. Speech synthesis based on hidden Markov models. *Proceedings of the IEEE*, 101(5): 1234–1252.

[20] Wang, W.; Xu, S.; Xu, B.; et al. 2016. First Step Towards End-to-End Parametric TTS Synthesis: Generating Spectral Parameters with Neural Attention. In *Interspeech*, 2243–2247.

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

[22] Xue, N. 2003. Chinese word segmentation as character tagging. In *International Journal of Computational Linguistics & Chinese Language Processing, Volume 8, Number 1, February 2003: Special Issue on Word Formation and Chinese Language Processing*, 29–48.

[23] Yamamoto, R.; Song, E.; and Kim, J.-M. 2020. Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6199–6203. IEEE.

[24] Yao, K.; and Zweig, G. 2015. Sequence-to-sequence neural net models for grapheme-to-phoneme conversion. *arXiv preprint arXiv:1506.00196*.

[25] Zen, H.; and Sak, H. 2015. Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4470–4474. IEEE.

[26] Zhang, J.; Pan, J.; Yin, X.; Li, C.; Liu, S.; Zhang, Y.; Wang, Y.; and Ma, Z. 2020. A hybrid text normalization system using multi-head self-attention for man-

darin. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6694–6698. IEEE.