INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

Real-time neural text-to-speech with sequence-to-sequence acoustic model and WaveGlow or single Gaussian WaveRNN vocoders

Takuma Okamoto[1]*, Tomoki Toda*[2,1]*, Yoshinori Shiga*[1]*, and Hisashi Kawai*[1]

[1]National Institute of Information and Communications Technology, Japan
[2]Information Technology Center, Nagoya University, Japan
{okamoto, yoshinori.shiga, hisashi.kawai}@nict.go.jp, tomoki@icts.nagoya-u.ac.jp

Abstract

This paper investigates real-time high-fidelity neural text-to-speech (TTS) systems. For real-time neural vocoders, WaveGlow is introduced and single Gaussian (SG) WaveRNN is proposed. The proposed SG-WaveRNN can predict continuous valued speech waveforms with half the synthesis time compared with vanilla WaveRNN with dual-softmax for 16 bit audio prediction. Additionally, a sequence-to-sequence (seq2seq) acoustic model (AM) for pitch accent languages, such as Japanese, is investigated by introducing Tacotron 2 architecture. In the seq2seq AM, full-context labels extracted from a text analyzer are used as input and they are directly converted into mel-spectrograms. The results of subjective experiment using a Japanese female corpus indicate that the proposed SG-WaveRNN vocoder with noise shaping can synthesize high-quality speech waveforms and real-time high-fidelity neural TTS systems can be realized with the seq2seq AM and WaveGlow or SG-WaveRNN vocoders. Especially, the seq2seq AM and WaveGlow vocoder conditioned on mel-spectrograms with simple PyTorch implementations can be realized with real-time factors 0.06 and 0.10 for inference using a GPU.

Index Terms: speech synthesis, neural vocoder, WaveGlow, WaveRNN, text-to-speech

1. Introduction

Real-time text-to-speech (TTS) techniques are among the most important speech communication technologies. Thanks to recent advances in deep learning, hidden Markov models (HMMs) have been replaced by deep neural networks (DNNs) [1] for the duration and acoustic models in statistical parametric speech synthesis (SPSS). Although conventional DNN-based SPSS systems with source-filter vocoders (for example, STRAIGHT [2]) can realize real-time synthesis [1, 3, 4], the synthesized speech quality is not very high. This is because of the pipeline structure, which separately trains the duration and acoustic models, and the introduction of source-filter vocoders.

To solve these problems, a neural network-based autoregressive (AR) generative model for raw audio, WaveNet, has been proposed [5]. Unlike conventional SPSS systems, WaveNet can directly synthesize speech waveforms from duration-predicted linguistic features with fundamental frequencies, and it outperforms conventional TTS systems [5] based on unit selection and SPSS. By focusing on the high performance of raw waveform modeling in WaveNet, neural vocoders that directly synthesize raw speech waveforms from acoustic features have been proposed [6, 7]; these outperform conventional source-filter vocoders in SPSS [8].

Additionally, such neural vocoders can realize end-to-end TTS, converting text to raw speech waveforms with sequence-to-sequence (seq2seq) neural networks [9–15]. Although conventional SPSS systems separately train duration and acoustic models, these seq2seq models can jointly train them simultaneously without a pipeline structure. By introducing a seq2seq model and a neural vocoder, to solve the pipeline structure and source-filter vocoder problems in conventional SPSS, the speech quality of English synthesized by Tacotron 2 can match that of natural speech [12]. In Tacotron 2, input characters are directly converted to mel-spectrograms with the seq2seq model, and speech waveforms are synthesized from the predicted mel-spectrograms by an AR WaveNet vocoder [12]. However, these seq2seq models cannot be used directly for pitch accent languages, such as Japanese [15, 16], and the synthesis speed of AR WaveNet is quite slow because the sequential synthesis of each sample requires substantial calculation time [5, 6].

Seq2seq acoustic models (AMs) for Japanese, with separately embedded phoneme and accentual-type sequences instead of characters, have been investigated [16]. However, these seq2seq AMs were found to be inferior to conventional pipeline models with full-context label input [16]. This result indicates the importance of full-context labels for pitch accent languages.

To overcome the synthesis speed problem in the AR WaveNet and SampleRNN neural vocoders [5, 6], two types of solutions have been proposed. The first type comprises AR models with simple structures, such as FFTNet [17], WaveRNN [18], and LPCNet [19]. In particular, WaveRNN and LPCNet can realize real-time synthesis using a mobile CPU by introducing a sparse gated recurrent unit (GRU). The other type of solution comprises flow-based [20] non-AR models that simultaneously generate all speech samples, such as parallel WaveNet [21–23] and WaveGlow [24]. Additionally, an alternative real-time approach, neural source-filter (NSF) [25], has been proposed. Real-time neural TTS systems with parallel WaveNet [21] and WaveRNN [18] have been realized, with duration-predicted linguistic features and fundamental frequencies, such as AR WaveNet TTS [5]. Additionally, WaveRNN can also be conditioned on mel-spectrograms [26].

To realize real-time high-fidelity neural TTS systems for pitch accent languages, this paper investigates a seq2seq AM for pitch accent languages and real-time neural vocoders for TTS. By focusing on the importance of full-context labels for pitch accent languages, a seq2seq AM with full-context label input is investigated. Unlike parallel WaveNet [21–23], based on the inverse autoregressive flow [27] and NSF, WaveGlow models can be trained directly, without teacher–student training and fundamental frequency analysis. Therefore, WaveGlow is introduced as a real-time neural vocoder. In addition, inspired by AR single Gaussian (SG) WaveNet [22] and FFTNet [23], SG-WaveRNN is proposed as a real-time AR neural vocoder; this is because it has potential for real-time synthesis with a mobile CPU if

Copyright © 2019 ISCA

http://dx.doi.org/10.21437/Interspeech.2019-1288

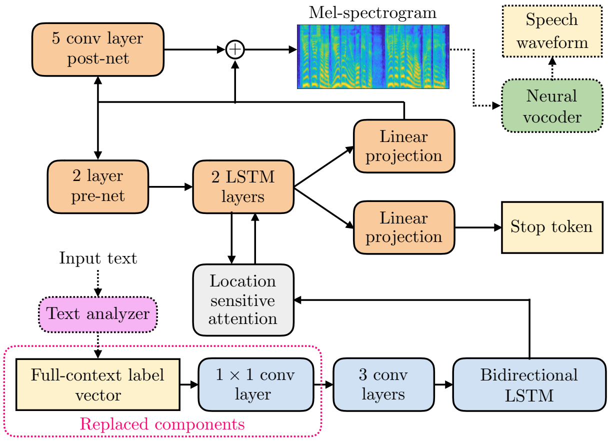


Figure 1: *Sequence-to-sequence acoustic model with full-context label input based on Tacotron 2 architecture.*

a sparse GRU can also be derived [18, 19], whereas WaveGlow requires a GPU. By integrating the seq2seq AM and neural vocoders, real-time neural TTS systems can be realized.

2. Sequence-to-sequence acoustic model with full-context label input

The seq2seq AMs for Japanese with separately embedded phoneme and accentual-type sequences are inferior to conventional pipeline models with full-context label input [16]. This indicates that only phoneme and accentual-type input is insufficient, and full-context label input might be important to seq2seq AMs for pitch accent languages. Therefore, a seq2seq AM with full-context label input, rather than phoneme and accentual-type sequences, is investigated, by extending the seq2seq architecture of Tacotron 2 [12].

Phoneme-level full-context labels are obtained as linguistic features from a text analyzer. In conventional pipeline TTS frameworks, duration models are first trained from the label vectors, and AMs predicting acoustic features for source-filter vocoders are then trained from the phoneme-level vectors or the HMM state-aligned frame-level vectors [3, 4]. In this paper, a seq2seq AM that predicts mel-spectrograms for neural vocoders is directly trained from the phoneme-level full-context label vectors. Full-context label vectors typically include past and future 2 contexts. As in [3], these past and future 2 contexts are also reduced in the seq2seq AM because it can access the past and future contexts through its bidirectional recurrent connections. The seq2seq AM architecture is almost the same as that of Tacotron 2 except for its input modules (Fig. 1). Unlike Tacotron 2, phoneme-level full-context label vectors extracted from a text analyzer are input to a 1×1 convolution layer instead of using character input and an embedding layer. The seq2seq AM is not an end-to-end framework but a language-independent framework, because phoneme-level full-context labels for all languages can be introduced directly.

3. Real-time neural vocoders for TTS

Neural vocoders for TTS are trained from the ground-truth mel-spectrograms in the training set, and speech waveforms for TTS are synthesized from mel-spectrograms predicted by the seq2seq AM with full-context label input.

3.1. WaveGlow

WaveGlow [24] is a promising deep generative model for raw audio that integrates a generative model for images, Glow [28], with WaveNet [5]. During training, an input speech waveform $\boldsymbol{x}$ is converted to Gaussian white noise $\boldsymbol{z}$. Conversely, a speech waveform is generated from Gaussian white noise by the inverse operations during the inference phase. By introducing the invertible 1×1 convolution and affine coupling layers, the loss function of the WaveGlow vocoder, with network parameters $\boldsymbol{\theta}$ conditioned on acoustic feature $\boldsymbol{h}$, is derived as

$$-\log p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \frac{\boldsymbol{z}(\boldsymbol{x})^T \boldsymbol{z}(\boldsymbol{x})}{2\sigma_{\mathrm{WG}}^2} - \sum_{j=0} \log \boldsymbol{s}_j(\boldsymbol{x}, \boldsymbol{h}) - \sum_{k=0} \log |\det(\boldsymbol{W}_k)|, \quad (1)$$

where $\boldsymbol{s}_j$, $\boldsymbol{W}_k$, and σ_{WG}^2 are the output coefficients of the jth WaveNet (corresponding to the standard deviation in SG-WaveNet [22, 23]) in the affine coupling layers, the kth weighting matrix of the invertible 1×1 convolution layers, and the assumed variance of the Gaussian distribution, respectively. Because of their sophisticated structure, WaveGlow models can be trained directly, without the need for teacher–student training, for parallel WaveNet [21–23], and all speech samples can be synthesized simultaneously, with acoustic features $\boldsymbol{h}$ and Gaussian white noise $\boldsymbol{z}$ [24].

3.2. Proposed single Gaussian WaveRNN

In vanilla WaveRNN, dual-softmax is introduced for predicting 16-bit raw audio, instead of the 8-bit μ-law coding defined in G.711 [29], which is used in WaveNet [5], FFTNet [17], and LPCNet [19], as shown in Fig. 2(a)[1]. However, two samplings are required to synthesize one audio sample in vanilla WaveRNN.

To reduce the number of times of sampling while predicting 16-bit raw audio waveforms, SG-WaveRNN is proposed, as shown in Fig. 2(b). As in AR SG-WaveNet [22] and FFTNet [23], the conditional distribution of $p(x_t|x_{<t}, \boldsymbol{\theta}; \boldsymbol{h})$ is defined as:

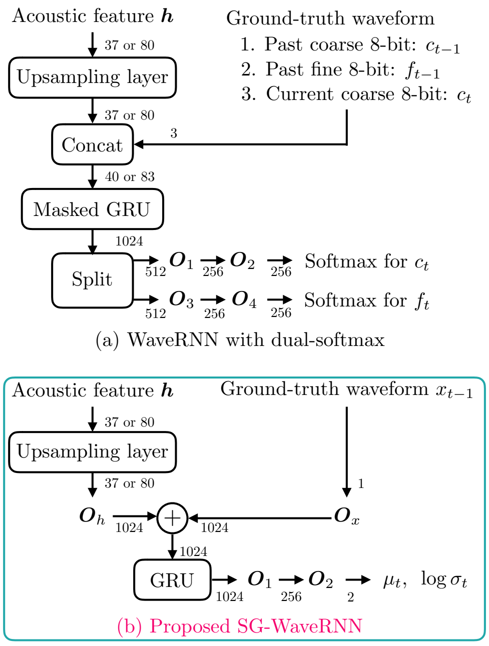
$$p(x_t|x_{<t}, \boldsymbol{\theta}; \boldsymbol{h}) = \mathcal{N}(\mu(x_{<t}; \boldsymbol{\theta}), \sigma(x_{<t}; \boldsymbol{\theta})), \quad (2)$$

where $\mu(x_{<t}; \boldsymbol{\theta})$ and $\sigma(x_{<t}; \boldsymbol{\theta})$ are the mean and standard deviation, respectively, predicted by the proposed SG-WaveRNN. Network parameters $\boldsymbol{\theta}$ are trained using maximum likelihood estimation [22] with the following loss function.

$$-\log p(x_t|x_{<t}) = \frac{1}{2}\log 2\pi + \frac{1}{2}\log \sigma_t^2 + \frac{(x_t - \mu_t)^2}{2\sigma_t^2}. \quad (3)$$

In the proposed SG-WaveRNN, additional feedforward layers $\boldsymbol{O}_h$ and $\boldsymbol{O}_x$ are introduced, like the conditioning structure in FFTNet [17]. By the proposed SG-WaveRNN, compared with vanilla WaveRNN, continuous-valued speech waveforms can be predicted with a simpler structure and half the synthesis time.

[1] The method of inputting conditioning vectors, such as linguistic or acoustic features, to a GRU layer is not disclosed in [18]. Therefore, in this paper, upsampled acoustic features are simply concatenated with coarse and fine audio vectors. A masked GRU (Fig. 2(a)) also includes a masked matrix, whereby the last coarse input c_t is only connected to the fine part of the states and only affects the fine output f_t, as in [18].



(a) WaveRNN with dual-softmax

(b) Proposed SG-WaveRNN

Figure 2: *WaveRNN training structure.*

3.3. Time-invariant noise shaping for neural vocoders

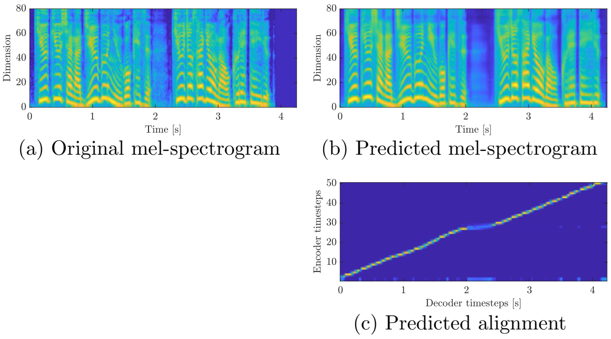
To reduce the adverse effects of noise signals caused by prediction errors in neural vocoders, time-invariant noise shaping based on perceptual weighting techniques can improve the synthesis quality of categorical and SG WaveNet and FFTNet vocoders [23, 30–32]. Therefore, this noise shaping method is also applied to the WaveGlow and SG-WaveRNN vocoders. In LPCNet, a simple linear predictive coding, instead of noise shaping, is applied for the same purpose [19].

4. Experiments

4.1. Experimental conditions

To evaluate the seq2seq AM with full-context label input, as well as the WaveGlow and proposed SG-WaveRNN vocoders, experiments were conducted using a Japanese female speech corpus with a sampling frequency of 24 kHz. 25,626 (about 22 h) and 20 utterances were used as the training set and test set, respectively. The WaveGlow and SG-WaveRNN vocoders were compared with the AR SG-WaveNet [22, 23], vanilla WaveRNN [18], and STRAIGHT [2] vocoders as references. As in [7, 23, 33], the SG-WaveNet, vanilla WaveRNN, and SG-WaveRNN vocoders, with simple acoustic features (SAF) constructed from the fundamental frequency and mel-cepstra [34, 35], were also evaluated for conventional pipeline SPSS systems. In the neural vocoders, both the analysis–synthesis (AS) and TTS conditions were evaluated. In the AS condition, the neural vocoders were trained from SAF or mel-spectrograms, and the AS waveforms were synthesized with the test sets' acoustic features. In the TTS condition, mel-spectrograms were predicted by the seq2seq AM with full-context label input, and the TTS waveforms were synthesized by the neural vocoders trained in the AS condition with the predicted mel-spectrograms.

Acoustic features $\boldsymbol{h}$ for SAF were analyzed every 5 ms over a Hann window with length 25 ms. Fundamental frequency f_{o},



(a) Original mel-spectrogram

(b) Predicted mel-spectrogram

(c) Predicted alignment

Figure 3: *Results of (a) original and (b) predicted mel-spectrograms and (c) predicted alignment for a test set.*

analyzed by an NDF algorithm [36], was used in all vocoders with SAF [23, 32, 33] and STRAIGHT. 35-dimensional mel-cepstra were analyzed from a simple short-time Fourier transform of windowed speech waveforms with warping coefficient $\alpha = 0.46$. In the neural vocoders with SAF, $(1+1+35=)$ 37-dimensional vectors constructed from continuous logarithmic f_{o}, voice/unvoice one-hot vector, and mel-cepstra (normalized to have a zero mean and unit variance) were used. In the neural vocoders with mel-spectrogram input, 80-dimensional log-mel-spectrograms were analyzed every 12.5 ms over a Hann window with length 85.3 ms, with a frequency band 125–7,600 Hz and normalized to the range of [0, 1], as in [12, 23].

Transposed convolution was also applied for the upsampling layers [5] and an Adam optimization algorithm was introduced in all of the neural network models. All neural network models were trained using four NVIDIA Tesla V100 GPUs.

In AR SG-WaveNet, the numbers of residual and skip channels were both set to 128. Twenty layers (10 dilations × 2 cycles) with a kernel size of two were used for the dilated causal convolution layers, as in [22, 23]. The learning rate, batch length, and batch size were 0.0002, 12,000 samples, and 8, respectively.

In vanilla WaveRNN and SG-WaveRNN, the numbers of input and output channels of the feedforward and GRU layers are described in Fig. 2. The learning rate, batch length, and batch size were 0.0001, 1,200 samples, and 256, respectively.

In WaveGlow, all of the network parameters were the same as used in [24]. The batch length and batch size were 16,000 samples and 8, respectively. As in [24], the learning rate was initially set to 0.0001 and reduced to 0.00005. In this paper, $\sigma_{\mathrm{WG}} = 1.0$ in Eq. (1) was used for both training and inference.

Like [23, 30–32], a mel-generalized cepstrum-based [34] noise shaping filtering, implemented by the mel-log spectrum approximation (MLSA) filter [35], was introduced, which could be calculated immediately by IIR filtering. A parameter to control noise energy in the formant regions was also set to 0.5.

In STRAIGHT, only the AS condition was evaluated. Both 35-dimensional mel-cepstra, with $\alpha = 0.46$ for the smooth vocal tract spectrum and aperiodicity components, were obtained from the original STRAIGHT spectrum and aperiodicity coefficients (1025 dimensions) and the vocoded waveforms were synthesized using the compressed mel-cepstra.

In TTS, the full-context labels were extracted by a text analyzer used in [37] and an actually implemented TTS system for VoiceTra[2]. Although the number of dimensions of the linguis-

[2] https://voicetra.nict.go.jp/en/index.html

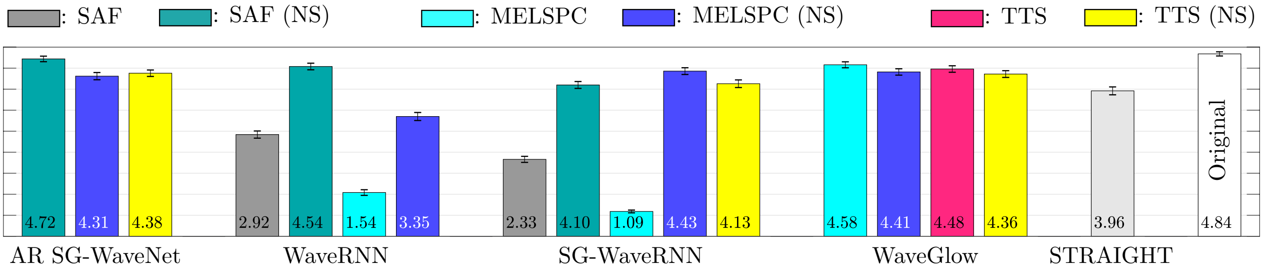


Figure 4: *Results of MOS test with 15 listening subjects. "SAF", "MELSPC", "TTS", and "NS" denote simple acoustic features, mel-spectrograms, text-to-speech, and noise shaping, respectively.*

tic feature vectors for a frame-wise DNN acoustic model was 483 [37], that for the seq2seq AM was 130 because the past and future 2 contexts were reduced, as described in Sec. 2. The label vectors were normalized to have a zero mean and unit variance. The number of output channels of the 1×1 convolution layer was 512. The model parameters of the seq2seq AM after the 3 convolution layers were the same as used in Tacotron 2 [12]. The learning rate and batch size were 0.001 and 32, respectively.

Figure 3 shows the results of the original and predicted mel-spectrograms, and the predicted encoder–decoder alignment for a test set. The results suggested that the alignment, as well as the mel-spectrograms, can be successfully trained by the seq2seq AM with full-context label input. The TTS speech waveforms were then synthesized by the neural vocoders with the mel-spectrograms predicted by the seq2seq AM.

4.2. Subjective evaluation

To subjectively evaluate the synthesized speech waveforms, mean opinion score (MOS) tests [38] were conducted[3]. All 20 utterances of the test set were used as the evaluation set. These were presented through headphones to 15 Japanese adult native speakers without hearing loss (20 utterances × 18 conditions including the original test set waveforms = 360 utterances).

The MOS results are plotted in Fig. 4. The synthesis qualities of the vanilla WaveRNN and proposed SG-WaveRNN vocoders, conditioned on both SAF and mel-spectrograms, were drastically improved by the noise shaping, although the training loss scores without noise shaping were lower than those with noise shaping. Conversely, the synthesis qualities for the WaveGlow vocoders with noise shaping were slightly degraded. This is because noise shaping can improve the robustness to sequential prediction errors in AR neural vocoders. Therefore, the noise shaping is only effective for AR neural vocoders. In particular, neural TTS systems with the seq2seq AM and AR SG-WaveNet, proposed SG-WaveRNN, and WaveGlow vocoders can successfully realize high synthesis quality, with MOS values over 4.0. Therefore, the seq2seq AM with full-context label input is effective for pitch accent languages.

4.3. Real-time factor evaluation

To evaluate the synthesis speed of the neural vocoders conditioned on mel-spectrograms, as well as the seq2seq AM, the real-time factors (RTFs) were measured, with an NVIDIA Tesla V100 GPU. In this paper, all modules were realized by simple PyTorch[4] implementations. The RTFs of the seq2seq AM and WaveGlow for inference were 0.06 and 0.10, respectively. In contrast, the RTFs of AR SG-WaveNet, vanilla WaveRNN, and SG-WaveRNN were 205, 15.4, and 8.3, respectively. The result was almost equivalent to that with simple TensorFlow[5] implementations in [18]. The proposed SG-WaveRNN can synthesize speech waveforms with little more than half of the inference time required by vanilla WaveRNN. To achieve real-time synthesis of SG-WaveRNN with a GPU, GPU kernel implementations (without PyTorch and TensorFlow) are required, as in [18, 19]; this is a focus of future work.

Consequently, a real-time high-fidelity neural TTS system can be realized by the seq2seq AM with full-context label input and the WaveGlow vocoder with an RTF of 0.16 by using a GPU, even with the use of a simple PyTorch implementation.

5. Future work

In this paper, neural vocoders trained from the ground-truth mel-spectrograms were used in TTS. To improve the synthesized speech quality in TTS, mel-spectrograms predicted by AMs should also be used for neural vocoder training, as with Tacotron 2 [12]. The WaveGlow and SG-WaveRNN neural vocoders should be compared with parallel WaveNet [21–23], LPCNet [19], and NSF [25]. To realize real-time synthesis for SG-WaveRNN, a GPU kernel implementation is required, as in [18, 19]. Additionally, sparse (and subscale) modifications for SG-WaveRNN should also be investigated for real-time synthesis with a mobile CPU, as with [18, 19]. Furthermore, the seq2seq AM with full-context label input should be compared with the conventional duration–acoustic pipeline models [3, 4, 8, 25], direct waveform generation methods with duration predicted linguistic features [5, 18, 21], and other seq2seq approaches with character or phoneme input [10–16] for other languages.

6. Conclusions

This paper provided real-time high-fidelity neural TTS systems using a seq2seq AM with full-context label input and the WaveGlow and SG-WaveRNN vocoders. The results of experiments indicated that the proposed SG-WaveRNN with noise shaping can synthesize high-quality speech Waveforms, and a real-time high-fidelity neural TTS system for Japanese can be realized by the seq2seq AM and the WaveGlow vocoder with an RTF of 0.16 using a GPU, even if a simple PyTorch implementation is used.

[3] Because TTS of vanilla WaveRNN with noise shaping cannot realize high-quality synthesis, it was not included in the MOS tests. The poor quality might be caused by the absence of feedforward layers for acoustic feature input.

[4] https://pytorch.org

[5] https://www.tensorflow.org

7. References

[1] Z.-H. Ling, S.-Y. Kang, H. Zen, A. Senior, M. Schuster, X.-J. Qian, H. Meng, and L. Deng, "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends," *IEEE Signal Process. Mag.*, vol. 32, no. 3, pp. 35–52, May 2015.

[2] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigné, "Restructuring speech representations using a pitch-adaptive timefrequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Commun.*, vol. 27, no. 3–4, pp. 187–207, Apr. 1999.

[3] H. Zen and H. Sak, "Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis," in *Proc. ICASSP*, Apr. 2015, pp. 4470–4474.

[4] Z. Wu, O. Watts, and S. King, "Merlin: An open source neural network speech synthesis system," in *Proc. SSW 9*, Sept. 2016, pp. 218–223.

[5] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," in *Proc. SSW 9*, Sept. 2016, p. 125.

[6] S. Mehri, K. Kumar, I. Gulrajani, R. Kumar, S. Jain, J. Sotelo, A. Courville, and Y. Bengio, "SampleRNN: An unconditional end-to-end neural audio generation model," in *Proc. ICLR*, Apr. 2017.

[7] A. Tamamori, T. Hayashi, K. Kobayashi, K. Takeda, and T. Toda, "Speaker-dependent WaveNet vocoder," in *Proc. Interspeech*, Aug. 2017, pp. 1118–1122.

[8] X. Wang, J. Lorenzo-Trueba, S. Takaki, L. Juvela, and J. Yamagishi, "A comparison of recent waveform generation and acoustic modeling methods for neural-network-based speech synthesis," in *Proc. ICASSP*, Apr. 2018, pp. 4804–4808.

[9] J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastner, A. Courville, and Y. Bengio, "Char2Wav: End-to-end speech synthesis," in *Proc. ICLR*, Apr. 2017.

[10] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech*, Aug. 2017, pp. 4006–4010.

[11] S. O. Arik, G. Diamos, A. Gibiansky, J. Miller, K. Peng, W. Ping, J. Raiman, and Y. Zhou, "Deep voice 2: Multi-speaker neural text-to-speech," in *Proc. NIPS*, Dec. 2017, pp. 2966–2974.

[12] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerry-Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," in *Proc. ICASSP*, Apr. 2018, pp. 4779–4783.

[13] W. Ping, K. Peng, A. Gibiansky, S. O. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep voice 3: Scaling text-to-speech with convolutional sequence learning," in *Proc. ICLR*, Apr. 2018.

[14] H. Tachibana, K. Uenoyama, and S. Aihara, "Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention," in *Proc. ICASSP*, Apr. 2018, pp. 4784–4788.

[15] J.-X. Zhang, Z.-H. Ling, and L.-R. Dai, "Forward attention in sequence-to-sequence acoustic modeling for speech synthesis," in *Proc. ICASSP*, Apr. 2018, pp. 4789–4739.

[16] Y. Yasuda, X. Wang, S. Takaki, and J. Yamagishi, "Investigation of enhanced Tacotron text-to-speech synthesis systems with self-attention for pitch accent language," in *Proc. ICASSP*, May 2019, pp. 6905–6909.

[17] Z. Jin, A. Finkelstein, G. J. Mysore, and J. Lu, "FFTNet: A real-time speaker-dependent neural vocoder," in *Proc. ICASSP*, Apr. 2018, pp. 2251–2255.

[18] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. van den Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," in *Proc. ICML*, July 2018, pp. 2415–2424.

[19] J.-M. Valin and J. Skoglund, "LPCNet: Improving neural speech synthesis through linear prediction," in *Proc. ICASSP*, May 2019, pp. 5826–7830.

[20] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *Proc. ICML*, July 2015, pp. 1530–1538.

[21] A. van den Oord, Y. Li, I. Babuschkin, K. Simonyan, O. Vinyals, K. Kavukcuoglu, G. van den Driessche, E. Lockhart, L. C. Cobo, F. Stimberg, N. Casagrande, D. Grewe, S. Noury, S. Dieleman, E. Elsen, N. Kalchbrenner, H. Zen, A. Graves, H. King, T. Walters, D. Belov, and D. Hassabis, "Parallel WaveNet: Fast high-fidelity speech synthesis," in *Proc. ICML*, July 2018, pp. 3915–3923.

[22] W. Ping, K. Peng, and J. Chen, "ClariNet: Parallel wave generation in end-to-end text-to-speech," in *Proc. ICLR*, May 2019.

[23] T. Okamoto, T. Toda, Y. Shiga, and H. Kawai, "Investigations of real-time Gaussian FFTNet and parallel WaveNet neural vocoders with simple acoustic features," in *Proc. ICASSP*, May 2019, pp. 7020–7024.

[24] R. Prenger, R. Valle, and B. Catanzaro, "WaveGlow: A flow-based generative network for speech synthesis," in *Proc. ICASSP*, May 2019, pp. 3617–3621.

[25] X. Wang, S. Takaki, and J. Yamagishi, "Neural source-filter-based waveform model for statistical parametric speech synthesis," in *Proc. ICASSP*, May 2019, pp. 5916–5920.

[26] B. Li, Y. Zhang, T. Sainath, Y. Wu, and W. Chan, "Bytes are all you need: End-to-end multilingual speech recognition and synthesis with bytes," in *Proc. ICASSP*, May 2019, pp. 5621–5625.

[27] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," in *Proc. NIPS*, Dec. 2016, pp. 4743–4751.

[28] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1×1 convolutions," in *Proc. NeurIPS*, Dec. 2018, pp. 10 215–10 224.

[29] I.-T. R. G. 711, *Pulse Code Modulation (PCM) of voice frequencies*, 1988.

[30] T. Hayashi, A. Tamamori, K. Kobayashi, K. Takeda, and T. Toda, "An investigation of multi-speaker training for WaveNet vocoder," in *Proc. ASRU*, Dec. 2017, pp. 712–718.

[31] K. Tachibana, T. Toda, Y. Shiga, and H. Kawai, "An investigation of noise shaping with perceptual weighting for WaveNet-based speech generation," in *Proc. ICASSP*, Apr. 2018, pp. 5664–5668.

[32] T. Okamoto, T. Toda, Y. Shiga, and H. Kawai, "Improving FFTNet vocoder with noise shaping and subband approaches," in *Proc. SLT*, Dec. 2018, pp. 304–311.

[33] T. Okamoto, K. Tachibana, T. Toda, Y. Shiga, and H. Kawai, "An investigation of subband WaveNet vocoder covering entire audible frequency range with limited acoustic features," in *Proc. ICASSP*, Apr. 2018, pp. 5654–5658.

[34] K. Tokuda, T. Kobayashi, T. Masuko, and S. Imai, "Mel-generalized cepstral analysis — A unified approach to speech spectral estimation," in *Proc. ICSLP*, Sept. 1994, pp. 1043–1046.

[35] T. Fukada, K. Tokuda, T. Kobayashi, and S. Imai, "An adaptive algorithm for mel-cepstral analysis of speech," in *Proc. ICASSP*, vol. 1, Mar. 1992, pp. 137–140.

[36] H. Kawahara, A. de Cheveigné, H. Banno, T. Takahashi, and T. Irino, "Nearly defect-free F0 trajectory extraction for expressive speech modifications based on STRAIGHT," in *Proc. Interspeech*, Sept. 2005, pp. 537–540.

[37] K. Tachibana, T. Toda, Y. Shiga, and H. Kawai, "Model integration for HMM- and DNN-based speech synthesis using product-of-experts framework," in *Proc. Interspeech*, Sept. 2016, pp. 2288–2292.

[38] I.-T. R. P. 800, *Methods for subjective determination of transmission quality*, 1996.