DNN based multi-speaker speech synthesis with temporal auxiliary speaker ID embedding

Junmo Lee, Kwangsub Song, Kyoungjin Noh, Tae-Jun Park, Joon-Hyuk Chang
Department of Electronic Engineering
Hanyang University
Seoul, South Korea
jchang@hanyang.ac.kr

Abstract— **In this paper, multi speaker speech synthesis using speaker embedding is proposed. The proposed model is based on Tacotron network, but post-processing network of the model is modified with dilated convolution layers, which used in Wavenet architecture, to make it more adaptive to speech. The model can generate multi speaker voice with only one neural network model by giving auxiliary input data, speaker embedding, to the network. This model shows successful result for generating two speaker's voices without significant deterioration of speech quality.**

Keywords—deep learning, sequence to sequence, speech synthesis, multi speaker speech synthesis

I. Introduction

Speech synthesis, also known as text to speech (TTS), is a technique for the generating speech data from the text data. The conventional speech synthesis mechanism is divided into three stages [1]: the text data analysis, acoustic model, and vocoder. At the text data analyzing stage, the text data converts to linguistic features that include durations, phones, etc. The recurrent neural network (RNN) or decision tree is commonly used for this stage. The acoustic model stage generates acoustic features based on linguistic features derived from the first stage of which the Gaussian mixture model and hidden Markov model (GMM-HMM) are commonly used for the acoustic models. The last stage is vocoder, which reconstructs raw audio from acoustic features.

Recently researchers presented speech synthesizers based on a neural network. To replace the first and second stages with the neural network, Tacotron [2] is proposed; for second and third stages, Wavenet [3] is proposed. For constructing an end to end neural speech synthesizer there is a main problem to overcome, a large difference between the length of the text data and the length of the speech data. Tacotron overcomes this problem by applying the sequence to sequence network [4] model with attention mechanism [5] to the synthesizing network. Wavenet overcomes this problem by using an external duration prediction model. Any duration prediction model, such as decision-tree or long short-term memory (LSTM) based models, makes it possible to expand the text data's length to equal that of the speech data. Both Tacotron and Wavenet models achieve a comparable mean opinion score (MOS) with conventional models. Rather than generating standardized speech data, techniques for generating more personalized speech data are becoming popular. Tecent research, such as Baidu's Deep Voice 2 [6], Deep Speaker [7], and emotional speech synthesizer [8] has been conducted to understand how additional data works.

In this paper, we propose a more generalized method to offer additional information as a condition for the multi-speaker speech synthesis. We adapt this method with speaker ID information to propose a multi-speaker speech synthesizer, that is a modified version of Tacotron, in which the input data is modified with auxiliary speaker ID data on each timestep. As for the single Tacotron network with this simple and effective modification, both a man and woman's speech data can be generated by changing the order without severe deterioration in speech quality.

II. Multi-speaker Speech Synthesizer

Tacotron is composed of a seq2seq with an attention mechanism and an additional encoding module called CBHG [2]. The CBHG module is used twice, once for text analysis, and again for speech parameter post processing.

Fig. 1 shows the block diagram of proposed model. For our model, the text analysis, specifically the input data of the whole system, is solely modified for multi-speaker function. It is different from an earlier conventional multi-speaker speech synthesis system that modified both the text analysis and acoustic model parts. Though giving information to the decoding part that is composed of an acoustic model and a post-processing module can be more helpful in generating speaker-separable speech data, it is insufficient because the decoding part is not fully controllable with the input data.

A. Speaker ID Embedding

The speaker ID label is embedded and concatenated with character embedding. For a multi-lingual translation task using the language data, embedding as the start token proved to be very efficient [9]; however, adding speaker ID information in the same way has several problems. Above all, unlike the translation model, our speech synthesis model uses monotonic attention [10]. With monotonic attention which forcibly pushes the attention along the encoder timestep,

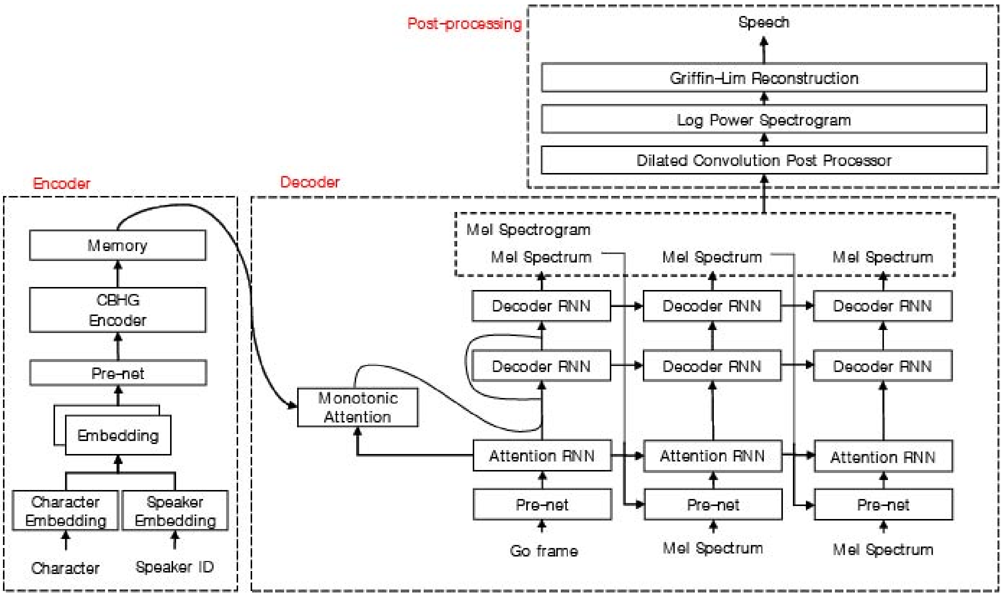

Fig. 1 Model structure for multi-speaker speech synthesis

giving the tag at the very front of the input data does not affect every decoder timestep sufficiently. In addition, word pieces of the translation model that are used as fundamental features of the input data are distinguishable among languages; however, in case of speech synthesis, character embedding remains the same. These differences hinder the model from training properly, which is proved by our experiment such that generated speech does not properly reflect the text data and speaker ID information.

To solve these problems, we propose speaker ID embedding to every position (i.e. timestep) of the input data. We thus modify by concatenating the input data with auxiliary speaker ID embedding as follows:

$$\mathbf{x}_s=[[\vec{x}_1, \vec{s}], [\vec{x}_2, \vec{s}]\ldots[\vec{x}_j, \vec{s}]\ldots [\vec{x}_n, \vec{s}]] \quad (1)$$

where $\vec{x}$ is the character embedding vector and $\vec{s}$ is the speaker ID embedding vector. A prime advantage of the proposed method is that it enables speaker ID embeddings to have an effect on every timestep. This process is the only modification for the multi-speaker speech synthesis of our model. The previous model is not able to show competitive speech quality, while our model generates speech with high quality, competitive to that of a single speaker model.

There are two main advantages that solely modify the encoder for multi speaker synthesis. First, the text data we use is raw text without any feature extraction except embedding. Since text embedding does not change the sequential information of the text data, it is fully controllable. For instance, when we want to add auxiliary information to the sentence, it is possible to add information at the desired position, but it is impossible to add information selectively to the decoder since decoder time steps do not have one to one correspondence with the text data. The following example of the input data shows the speaker ID changing in single utterance:

$$\mathbf{x_s} = [[\vec{x}_1 , \vec{s}_1], [\vec{x}_2, \vec{s}_1] \ldots [\vec{x}_j, \vec{s}_1], [\vec{x}_{j+1}, \vec{s}_2] \ldots [\vec{x}_n, \vec{s}_2]] \quad (2)$$

Speaker ID information changes at timestep j from s_1 to s_2. As a result, generated speech also undergoes speaker change in the middle of the sentence. But the point at which the speaker is changing is not equal to the input data. The speech is a little later than the input; in several examples, the two speakers' speech were not clearly separated.

B. Wavenet Postprocessor

There is a speech quality problem caused by lack of the data. Compared to the original Tacotron that trained with 21 hours of data, our model trained with only 7 hours of data for each speaker. To enhance speech quality, we modify Tacotron's post processing CBHG. CBHG, especially the convolution bank, is a good model for the sequence data, such as text; however, for speech data, a wider receptive field is often required. We employ the Wavenet's dilated convolution layer architecture [3], which enables us to obtain a very wide receptive field with relatively low computational cost.

In addition to changing the post processing network, monotonic attention helps the model easily form attention alignment. Even though monotonic attention is used, it is difficult to form clear attention alignment when we generate fewer than three mel spectrogram for single decoder timestep. Rather than simply adding two losses, multiplying the coefficient to post processing loss and adding the result with seq2seq loss produces clear attention alignment. The final loss is given as follows:

$$L_{total}=L_{seq2seq}+ \alpha*L_{post\ processing} \quad (3)$$

where α varies from 0.5-1.0. The compensation for post processing prevents the model from only relying on post processing network and not the encoder network. In addition to loss compensation, scheduled sampling also helps the model achieve firm attention alignments, but the quality of speech generated declines.

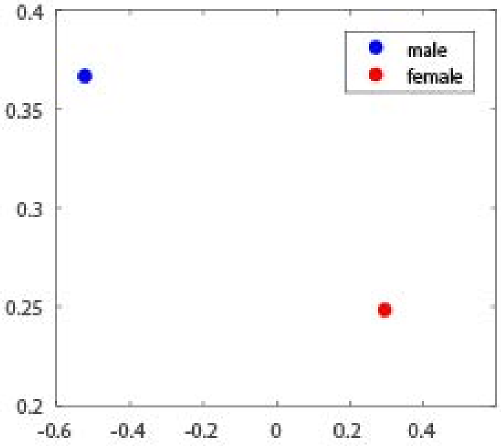

Fig. 2. Speaker ID embedding of two speakers

III. EXPERIMENTS

The multi-speaker Tacotron was trained with Korean speech synthesis dataset that contains

<text, audio, speaker ID> sets. The dataset has two kinds of text script: a spell transcription and a phonetic transcription. The model trained with the phonetic transcription produces better results, but the experiments we conducted used the spell transcription. The dataset has two speakers, one male and one female. The speakers' text scripts were the same. The dataset was approximately 7 hours for each speaker, which was considerably less than the original Tacotron experiment. The generated samples were comparable to those of the original but a bit noisier. While generating samples, the more unfamiliar was the word, the more inaccurate was the pronunciation. The generated samples are available at Github:

https://github.com/ljun4121/Multi-speaker_speech_synthesis

A. Speaker ID Embedding

While training the multi speaker Tacotron, we were most interested in how the speaker ID data was reflected in the speech synthesizer. Fig. 2 shows a 2-dimensional principal component analysis (PCA) of the speaker ID embedding we trained. The two speakers in dataset were separated in vector space. We synthesized several samples to analyze the performance of speaker ID embedding.

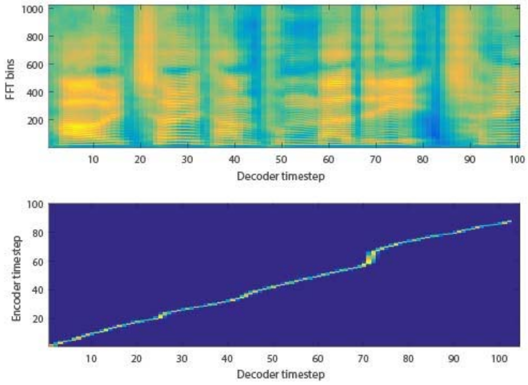

Fig. 5. Female speaker sample spectrogram and attention alignment

Fig. 3 and Fig. 4 show the attention alignments and spectrogram images of the generated samples. Both male and female speakers' attention alignments were very clearly formed.

We aimed at generating samples with an embedding vector which locate in the middle of the two speaker IDs' embedding:

$$s_{\text{middle}}=\frac{s_1+s_2}{2} \tag{4}$$

Fig. 5 exhibits the attention alignment and spectrogram of the generated sample with to s_{middle}. The sample continuously shifted between the two speakers. Speaker shifting was the most severe at the middle of the two speaker ID embeddings. With several generations of various embedding points, a plane dividing embedding vector space is assumed, and near the plane, the speaker confusion is intensified. It is inferred that the embedding vector acts as a selector, and the acoustic model composed of decoder RNNs is formed discontinuously and independently.

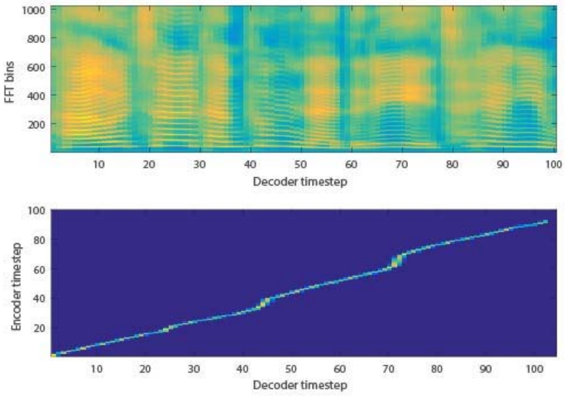

Fig. 4. Mixed speaker sample spectrogram and attention alignment

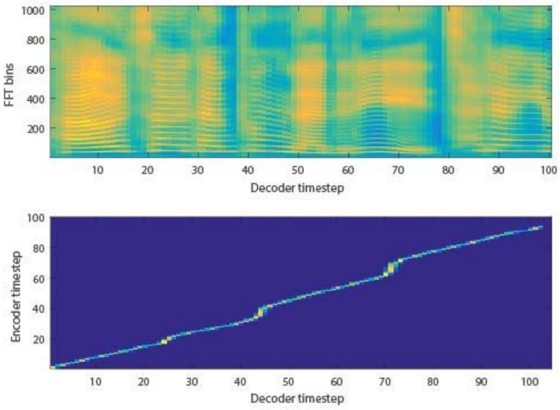

Fig. 3. male speaker sample spectrogram and attention alignment

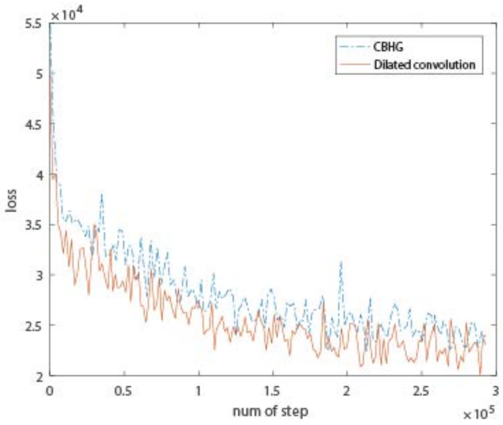

Fig. 6. Loss comparison between CBHG and dilated convolution

B. *Wavenet Postprocessor*

Fig. 6 depicts the loss comparison between CBHG postprocessor and dilated convolution postprocessor. The CBHG module is the same with the original tacotron's. The dilated convolution postprocessor is the same with Wavenet's dilated convolution layers. Dilation coefficients we used is as follows:

$$[1, 2, 4, 8, 16, 32, 64, \\ 1, 2, 4, 8, 16, 32, 64, \\ 1, 2, 4, 8, 16, 32, 64]$$

With the dilation coefficients, training time per step was about the same with CBHG postprocessor. The loss of model using dilated convolution layers is definitely lower, however in terms of speech quality the subtle noise that we want to remove was not significantly decreased.

C. *discussion*

We have proposed multi speaker Tacotron that takes combined text information and speaker information as an input and generates speaker separable speech data. We have proposed generalized method to control speech data generation with certain additional information. We have applied the method to multi speaker embedding and have figured out what's happening in the middle of embedding data. In our experiment, the acoustic models of two speakers are formed discontinuously, and significant amount of data for each speaker was needed to train. For constructing personalized speech synthesizer, to make a synthesizer model less data consuming is crucial. We are investigating designing a more efficient acoustic model. Furthermore, to apply our generalized method to give additional information with other data rather than speaker is in progress.

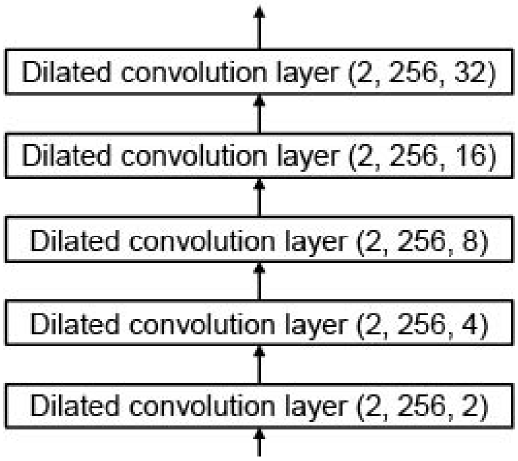
Dilated convolution layer (2, 256, 32)
Dilated convolution layer (2, 256, 16)
Dilated convolution layer (2, 256, 8)
Dilated convolution layer (2, 256, 4)
Dilated convolution layer (2, 256, 2)

Fig. 7. Dilation block (kernel size, filter size, dilation coefficient)

ACKNOWLEDGMENT

This work was supported by Institute for Information \& communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2017-0-00474, Intelligent Signal Processing for AI Speaker Voice Guardian)

This research was supported by Projects for Research and Development of Police science and Technology under Center for Research and Development of Police science and Technology and Korean National Police Agency funded by the Ministry of Science and ICT(PA-J000001-2017-101).

REFERENCES

[1] Zen, Heiga, Keiichi Tokuda, and Alan W. Black. "Statistical parametric speech synthesis." Speech Communication 51.11 (2009): 1039-1064.

[2] Wang, Yuxuan, R. J. Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J. Weiss, Navdeep Jaitly, Zongheng Yang et al. "Tacotron: A fully end-to-end text-to-speech synthesis model." arXiv preprint (2017).

[3] Van Den Oord, Aäron, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu. "WaveNet: A generative model for raw audio." In SSW, p. 125. 2016.

[4] Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le. "Sequence to sequence learning with neural networks." In Advances in neural information processing systems, pp. 3104-3112. 2014.

[5] Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate." arXiv preprint arXiv:1409.0473 (2014).

[6] Arik, Sercan, Gregory Diamos, Andrew Gibiansky, John Miller, Kainan Peng, Wei Ping, Jonathan Raiman, and Yanqi Zhou. "Deep voice 2: Multi-speaker neural text-to-speech." arXiv preprint arXiv:1705.08947 (2017).

[7] Li, Chao, Xiaokong Ma, Bing Jiang, Xiangang Li, Xuewei Zhang, Xiao Liu, Ying Cao, Ajay Kannan, and Zhenyao Zhu. "Deep speaker: an end-to-end neural speaker embedding system." arXiv preprint arXiv:1705.02304 (2017).

[8] Lee, Younggun, Azam Rabiee, and Soo-Young Lee. "Emotional End-to-End Neural Speech Synthesizer." arXiv preprint arXiv:1711.05447 (2017).

[9] Wu, Yonghui, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun et al. "Google's neural machine translation system: Bridging the gap between human and machine translation." arXiv preprint arXiv:1609.08144 (2016).

[10] Luong, Minh-Thang, Hieu Pham, and Christopher D. Manning. "Effective approaches to attention-based neural machine translation." arXiv preprint arXiv:1508.04025 (2015).