INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Phoneme Duration Modeling Using Speech Rhythm-Based Speaker Embeddings for Multi-Speaker Speech Synthesis

Kenichi Fujita, Atsushi Ando, Yusuke Ijima

NTT Corporation, Japan

kenichi.fujita.wv@hco.ntt.co.jp

Abstract

This paper proposes a novel speech-rhythm-based method for speaker embeddings. Conventionally spectral feature-based speaker embedding vectors such as the x-vector are used as auxiliary information for multi-speaker speech synthesis. However, speech synthesis with conventional embeddings has difficulty reproducing the target speaker's speech rhythm, one of the important factors among speaker characteristics, because spectral features do not explicitly include speech rhythm. In this paper, speaker embeddings that take speech rhythm information into account are introduced to achieve phoneme duration modeling using a few utterances by the target speaker. A novel point of the proposed method is that rhythm-based embeddings are extracted with phonemes and their durations. They are extracted with a speaker identification model similar to the conventional spectral feature-based one. We conducted two experiments: speaker embeddings generation and speech synthesis with generated embeddings. We show that the proposed model has an EER of 10.3% in speaker identification even with only speech rhythm. Visualizing the embeddings shows that utterances with similar rhythms are also similar in their speaker embeddings. The results of an objective and subjective evaluation on speech synthesis demonstrate that the proposed method can synthesize speech with speech rhythm closer to the target speaker.

Index Terms: speaker embedding, phoneme duration, speech synthesis, speaking rhythm

1. Introduction

Recently, Deep Neural Network (DNN) methods such as DNN speech synthesis [1], end-to-end speech synthesis [2], and neural waveform generation [3, 4, 5] have been used for speech synthesis. These methods generally require a large amount of speech data uttered by a single speaker. Methods have been proposed that use multiple-speaker speech data to achieve higher quality speech synthesis [6, 7, 8]. The advantages of multi-speaker speech synthesis are, for example, the reproduction of the speakers' characteristics with a smaller number of utterances from target speakers [6, 7, 8], and stable parameter generation in end-to-end speech synthesis [9].

In DNN-based speech synthesis with multi-speaker speech data, speaker characteristics are utilized as auxiliary information in addition to linguistic information (Fig. 1). Some studies use speaker codes, onehot vectors corresponding to each speaker, as the auxiliary input [10, 11]. Methods that use speaker codes improve the naturalness of the synthesized speech but are only applicable for speakers appearing in training data. Others use speaker embedding vectors such as the i-vector [12], d-vector [13], and x-vector [14] to synthesize unseen speakers in the training data [15, 16]. For example, Wu *et al.* applied the i-vector [15], and Doddipatla *et al.* applied the d-vector [16] for speaker adaptation. These studies showed

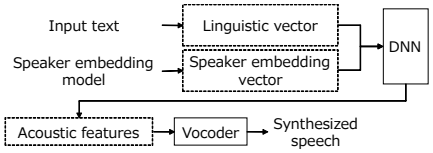


Figure 1: *Overview of multi-speaker speech synthesis.*

that methods with speaker embeddings [7, 17] accurately reproduced individual utterance features to some extent. However, since these methods mainly focused on modeling acoustic features (spectral features and F0), the synthesized speech would not reflect individual speech rhythm, one of the important factors among speaker characteristics. One reason for this is that the conventional models for extracting speaker embeddings trained from spectral features do not explicitly take speech rhythm information into account. Therefore, if speaker embedding vectors considering speech rhythm were obtained, a higher reproduction quality for the target speaker's speech synthesis would be achieved.

To this end, we propose a speech-rhythm-based speaker embedding extraction method. The key idea of the proposed method is the use of only rhythm-based features, i.e., phonemes and their durations, to construct a speaker identification DNN. If speaker identification could be achieved by using only phonemes and their durations, the obtained embedding vector, the output of a bottleneck layer, could capture speaker characteristics regarding speech rhythm. Two experiments, speaker identification and speech synthesis, were conducted to evaluate the proposed rhythm-based speaker embeddings. The former shows that the proposed embeddings achieved an equal error rate (EER) of 10.3% and were distributed within a close space when speakers had close speech rhythms. The latter shows that synthesized speech with the proposed embeddings outperformed that with the conventional embeddings (x-vector) in both objective and subjective evaluations.

2. Speaker embedding method

2.1. Conventional method

This section describes conventional speaker embeddings based on the x-vector [14]. A speaker identification model is used to extract the x-vector from acoustic features as described in Fig. 2(a). The input acoustic features are processed by a model composed mainly of three blocks. The first block is the DNN block composed of a time-delay neural network (TDNN) [18] that extracts frame-level features. The second is an attention block that converts frame-level features into a fixed-dimensional vector. The third is a fully connected layer block that extracts utterance-level features. For training, either speaker classification loss (e.g., softmax) or metric learning loss (e.g., angular prototypical [19]) is used. The speaker embedding vector is extracted from a particular layer in the third block. The

http://dx.doi.org/10.21437/Interspeech.2021-826

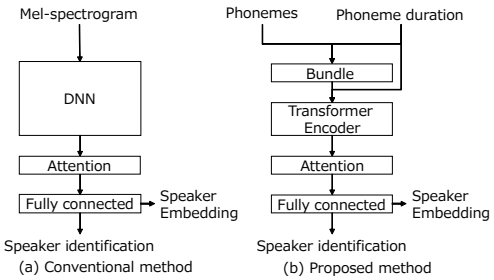


Figure 2: *Comparison of conventional and proposed methods.*

vectors obtained from this method are not explicitly based on speech-rhythm characteristics.

2.2. Proposed rhythm-based speaker embeddings

This section proposes the rhythm-based speaker embedding extraction. The key idea of the proposed method is to leverage phonemes and their durations in speaker identification to embed the speech rhythm of each speaker. Pairs of phonemes and their durations are known as representing speech rhythm. The phonemes are given as onehot vectors and the phoneme durations are given as one dimension vectors. As input features, these are concatenated and transformed into appropriate dimension feature vectors.

To extract features from a phoneme-based sequence, we changed the conventional model structure up to the attention block. Fig. 2(b) shows the detailed structure of the proposed model. The DNN block is replaced with a bundle block and transformer encoder, and the attention block is replaced with temporal direction attention from feature-dimension direction attention. The bundle block concatenates several preceding and following input feature vectors (N_{pre}, N_{follow}: respectively the number of preceding and following concatenated features) to extract local features. The transformer encoder block [20], a block with several stacked blocks of self-attention and ResNet on top of each other, extracts the features of the series. Conventional models extract comparably local features by using a TDNN at the layers before the attention block. However, the speech rhythm feature would derive from a longer speech context. Therefore, we changed the TDNN to the transformer encoder to extract features by looking at a whole sentence.

3. Experiments

We conducted two experiments. The first one evaluated the speaker identification performance and visualized the distribution of speaker embeddings to investigate the characteristics of rhythm-based speaker embeddings. We next conducted objective and subjective evaluations to investigate the performance in phoneme duration prediction.

3.1. Experiment for extracting speaker embedding

The goal of this experiment was to evaluate speaker individuality in pairs of phonemes and their duration through speaker identification. If a pair has features of each speaker, the speaker identification will show at least a moderate score. Therefore, the proposed vector may lead to a lower performance than the conventional vectors designed for speaker identification.

3.1.1. Dataset

The experiment used a Japanese speech database containing 492 speakers. This database consists of several speaker types including professional speakers, i.e., newscasters, narrators, and voice actors, non-professional speakers, and L2 speaker. Since this dataset was constructed for speech synthesis, the nature is different from that of spontaneous speech. Each speaker was instructed to maintain a constant speaking rate and tone during the recording. Furthermore, about 300 of the 492 speakers uttered the same script for speech. 114,355 utterances by 470 speakers were used for training, and 8,137 utterances by 22 speakers were used for evaluation. In the evaluation set, the number of the utterance pairs consisted of the same speakers and that with the different speakers are the same. The phoneme duration of each utterance was obtained from the phoneme boundary information manually segmented to each utterance.

Table 1: *Speaker identification results for each condition.*

method	input feature	EER [%]
x-vector	mel-spectrogram	2.0
proposed	phonemes, durations	10.3
phonemes only	phonemes	17.1
(chance rate)	-	50.0

3.1.2. Training conditions

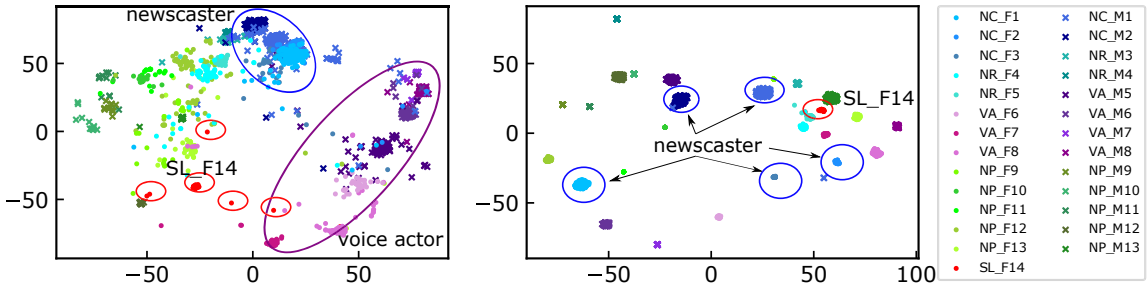
We trained the conventional x-vector and the proposed models. The structure of the proposed model is described in section 2.2. The input features were 56-dim onehot phoneme vectors and the phoneme duration. For the training, we utilized angular prototypical loss, a metric learning loss. The model was trained up to 1000 epochs and evaluated every 10 epoch with the equal error rate (EER). The training was stopped when the EER showed the lowest value in the evaluation set. The network was a two-layer transformer encoder that was proceeded by the bundle block and followed by the attention block and fully connected block. The number of hidden nodes in each hidden layer was 300. In the bundle block, the number of preceding features was $N_{pre} = 2$ and that of following features was $N_{follow} = 2$. The transformer encoder had 2 layers, 64 units, and 8 heads. The attention layer aggregates phoneme-level features into 32 dim feature vectors by using a self-attentive structure with 64 units and 8 heads. The fully connected block output 32-dim bottleneck features. These parameters were determined by preliminary experiments. To evaluate the contribution of phoneme duration to speaker-characteristic extraction, we also trained the model with only onehot phoneme vectors.

The conventional model (x-vector) is trained from mel-spectrograms. The model structure is Fast ResNet-34, which consists of a convolution layer and multi-stage ResNet layers [19]. The input was 40-dim mel-spectrograms extracted by mel-spectrogram analysis computed every 10ms with a 25ms Hamming window from 16 kHz speech signals. For the attention layer, we used self-attentive pooling (SAP) [21]. As in the proposed method, the dimension of the speaker embedding vector was set to 32, and we used angular prototypical loss.

We call the conventional model "x-vector", the proposed model trained with phonemes and their durations "proposed", and that with only phonemes "phonemes only."

3.1.3. Performance evaluation of speaker identification

Table 1 shows the speaker identification performance of each model. The EER of the proposed model was 10.3%. This indicates that the performance of the proposed method was moderate with only phonemes and their durations. Note that the phonemes only model had an EER of 17.1%, which indicates



(a) *Proposed embeddings.* (b) *x – vectors.*

Figure 3: *Distribution of speaker embedding vectors using proposed method. The blue, purple, and red circles respectively indicate newscasters, voice actors, and an L2 speaker.*

that some speech in the dataset was phonetically biased because 300 speakers in the dataset uttered the same script. Compared with the performance of the x-vector, that of the proposed method was lower. However, the result implies pairs of phonemes and their durations are related to speaker characteristics. If used with spectral features, the speaker recognition score may also improve, although this is out of the target of this study.

3.1.4. Visualization of speaker embedding vector

Next, we visualized the extracted speaker embedding vectors with t-SNE [22] to evaluate the spatial distribution of them. Figures 3(a) and (b) respectively show distributions of the proposed embedding vectors and the x-vector. Females and males are represented as "F" with a circle marker and "M" with a cross marker. In the graph legend, the caption of each speaker shows the following properties: "NP" for nonprofessional speakers, "NR" for reading style of narrators, "VA" for character-acting style of voice actors, "NC" for news reading style of newscasters, and "SL" for an L2 speaker.

Figure 3(a) shows that speakers with closer speech rhythms existed in a closer area in the space. For example, the vectors from the utterances of newscasters, character voice actors, and the L2 speaker demonstrates this tendency. The newscasters (NC_F1, NC_F2, NC_F3, NC_M1, NC_M2) and the character voice actors (VA_F6, VA_F7, VA_F8, VA_M5, VA_M6, VA_M7, VA_M8), whose speech rhythms were close with each other, were respectively clustered close together in the space (the blue and purple circles in the figure). The L2 speaker SL_F14, whose utterance was unsteady and whose rhythm changed from utterance to utterance, was widely distributed (the area surrounded by red circles in the figure).

The distribution of the x-vector (Fig. 3(b)), which is based on spectral features, shows different tendencies from those of the proposed model. For example, we can see that newscasters (NC_F1, NC_F2, NC_F3, NC_M1, NC_M2) were not located close to each other in Fig. 3(b). In addition, SL_F14, utterances by the L2 speaker, were not spread out widely in the space but spreads similarly to the other speakers. These results indicates that the proposed vector extracts features based on speech rhythm not reflected enough in the x-vector.

3.2. Experiment for phoneme duration prediction

3.2.1. Dataset

The same database explained in section 3.1.1 was used for the phoneme duration prediction. The database was separated into three groups: Train-A, Train-B, and Eval. These three groups were for the speech synthesis with closed speaker data and open speaker data. Train-A was used for training the open-speaker case, and both Train-A and Train-B were used together for the closed-speaker case. Train-A consisted of about 80,000 utterances by 470 speakers, which was 80% of the training data of the original database. Train-B and Eval were from the evaluation data. Train-B, training data for the evaluation-data speakers, consisted of 5 utterances from each of 22 evaluation-data speakers extracted from the evaluation data. We used only five utterances for the evaluation-data speakers (Train-B). This is to evaluate the model trained with a few utterances. Eval was the 20% of the evaluation data (about 1,600 utterances) except for the utterances in Train-B.

Table 2: *Training dataset for phoneme duration prediction.*

	proposed	onehot	x-vector
Train-A	✓	✓	✓
Train-B	×	✓	×

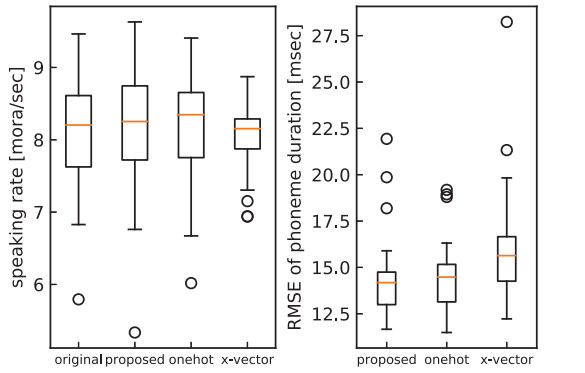
3.2.2. Training conditions

As the model for predicting phoneme duration, we adopted an explicit duration prediction model in the same manner as DNN-based speech synthesis [1] and a non-autoregressive speech synthesis model, e.g., FastSpeeech2 [23]. This is because directly evaluating the predicted duration with sequence-to-sequence model, e.g., Tacotron2 [2], is difficult.

Phoneme duration prediction models consist of 6 transformer encoder blocks. Each block has a dimension of 64 with 8 heads. The input is a 303-dimensional linguistic vector and the speaker embedding vector. The output is the one-dimensional phoneme duration of each phoneme. As the embedding vector of each speaker, we used the average values from embedding vectors of all utterances obtained from Train-A and Train-B (the embedding vectors of the speakers in Train-B were calculated from five utterances included in Train-B). For the training, we used mean squared error (MSE) loss. The model was trained up to 1000 epochs and stopped when the MSE of the evaluation set showed the lowest score.

We trained each model with the three types of vectors: the proposed embedding vector, onehot vector, and x-vector. They were trained with the datasets shown in Table 2. As shown in the table, the proposed and x-vector methods were trained with the open speaker case. In contrast, the onehot method is trained with the closed-speaker case. We trained the onehot model as an ideal case where retraining of the duration prediction model with the target speaker's utterances is allowed.

The speech parameters were generated from each predicted phoneme duration on the basis of multi-speaker DNN-based speech synthesis using onehot speaker code [6]. To evaluate the difference in phoneme duration predicted by each method,



(a) *Speaking rate* (b) *RMSE of phoneme duration*

Figure 4: *Objective evaluations for each condition.*

the same DNN acoustic model was used for speech parameter generation. We used two linear layers and two unidirectional LSTM layers. The number of units per layer was 512. We used ReLU as the activation function. The acoustic feature vector consisted of 40 mel-cepstral coefficients including the zeroth coefficient, log F0, voice/unvoiced flag, and 10-band aperiodicity values extracted by STRAIGHT [24]. The linguistic vector consisted of 307-dimensional vectors containing phoneme and accentual information.

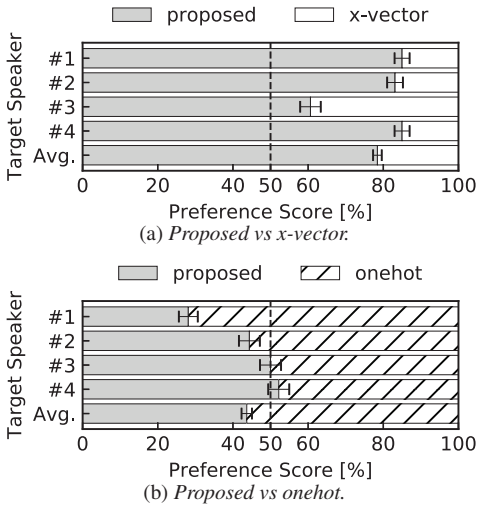
3.2.3. Objective evaluation

To evaluate the effectiveness of the proposed method, we first compared the distribution of speaking rates obtained from each method. Figure 4(a) shows a box plot of the speaking rate from the 22 evaluation-data speakers. We can see that the distribution of the original speech was wide, and that of the proposed and onehot methods had a similar tendency to the original speech. In contrast, that of the x-vector was narrower than the other three distributions. These results indicate that the proposed speaker embedding vector can capture speaker characteristics regarding speech rhythm and the x-vector cannot.

Next, we evaluated the performance of the phoneme duration prediction. Figure 4(b) shows a box plot of the root mean square error (RMSE) for each condition. It can be seen that the RMSEs of the proposed model and onehot model were almost the same. Note that the 22 evaluation-data speakers were open speakers for the proposed model, and were closed speakers for the onehot one. In other words, the onehot model requires retraining with evaluation-data speakers, and this retraining is not required for the proposed model. It is indicated that the proposed model predicted the phoneme durations of the open speaker as accurately as the onehot model trained with open speakers' utterances. This is an advantage of the proposed method because retraining for speakers not in the training data requires data for adequate adaptation as well as computational time and resources. Second, the RMSE of the x-vector is worse than that of the proposed method. This indicates that the proposed speaker-embedding vector can capture speech rhythm better compared with the x-vector.

3.2.4. Subjective Evaluation

To evaluate the performance of the proposed method, we conducted XAB listening tests. All permutations of synthetic speech pairs were presented in two orders (XAB and XBA), to eliminate bias in the order of stimuli. To evaluate only the difference in phoneme duration, the reference speech was the



(a) *Proposed vs x-vector.*

(b) *Proposed vs onehot.*

Figure 5: *Preference score from subjective evaluation. Error bars show 95% confidence intervals.*

synthetic speech from the same DNN acoustic model with the original phoneme duration of the target speaker. The subjects were four males and four females. Each was presented with synthesized speech samples and then asked which sample was similar to the reference speech. 20 sentences, each from four speakers, were used for the evaluation. The four speakers were selected from the 22 evaluation-data speakers sorted by RMSE (Fig. 4(b)) at equal intervals.

Figure 5 shows the preference scores for each target speaker. The proposed vector had better performance than the x-vector, and slightly worse performance than the onehot vector. This indicates that the proposed method makes it possible to synthesize speech closer to that target speaker than the spectral feature-based x-vector. Compared with the onehot model, the proposed method also achieved almost the same quality except for target speaker #1. Since speaker #1 has an outlier speech rhythm (shown in Fig. 4(a)), particularly slow speech rhythm, in the dataset, the retrained onehot model effectively reproduced speaker #1. To overcome this problem, we will explore data augmentation methods, one promising approach to reproducing speakers with outlier characteristics [25].

4. Conclusions

In this study, we proposed a method of extracting speaker embedding vectors capturing speech-rhythm features using phonemes and their durations as input. Conventionally, speaker embedding vectors have mainly been based on acoustic features and not explicitly focused on speech rhythm, an important factor in speaker characteristics. We considered phonemes and their durations to represent speech rhythm and used them as input for the speaker identification model. The performance of the proposed vectors was moderate in speaker identification, and vectors from close-speech-rhythm utterances were projected to a close area in the proposed speaker embedding space. It was also confirmed that our proposed method is applicable for better modeling of phoneme duration. Future work includes the application of other recent DNN speech synthesis methods such as the duration predictor in FastSpeech2 [23] and AlignTTS [26], as well as higher quality end-to-end speech synthesis (e.g., Tacotron2 [2]) inputting both the proposed and x-vectors as auxiliary information.

5. References

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

[2] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.

[3] 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," *arXiv e-prints*, p. arXiv:1609.03499, Sep. 2016.

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

[5] J.-M. Valin and J. Skoglund, "LPCNet: Improving neural speech synthesis through linear prediction," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5891–5895.

[6] N. Hojo, Y. Ijima, and H. Mizuno, "DNN-based speech synthesis using speaker codes," *IEICE TRANSACTIONS on Information and Systems*, vol. 101, no. 2, pp. 462–472, 2018.

[7] E. Cooper, C.-I. Lai, Y. Yasuda, F. Fang, X. Wang, N. Chen, and J. Yamagishi, "Zero-shot multi-speaker text-to-speech with state-of-the-art neural speaker embeddings," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6184–6188.

[8] Y. Fan, Y. Qian, F. K. Soong, and L. He, "Multi-speaker modeling and speaker adaptation for DNN-based TTS synthesis," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 4475–4479.

[9] J. Latorre, J. Lachowicz, J. Lorenzo-Trueba, T. Merritt, T. Drugman, S. Ronanki, and V. Klimkov, "Effect of data reduction on sequence-to-sequence neural TTS," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 7075–7079.

[10] H.-T. Luong and J. Yamagishi, "Scaling and bias codes for modeling speaker-adaptive DNN-based speech synthesis systems," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 610–617.

[11] H.-T. Luong, X. Wang, J. Yamagishi, and N. Nishizawa, "Training multi-speaker neural text-to-speech systems using speaker-imbalanced speech corpora," in *INTERSPEECH 2019*, 2019, pp. 1303–1307.

[12] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2010.

[13] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5115–5119.

[14] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[15] Z. Wu, P. Swietojanski, C. Veaux, S. Renals, and S. King, "A study of speaker adaptation for DNN-based speech synthesis," in *INTERSPEECH 2015*, 2015, pp. 879–883.

[16] R. Doddipatla, N. Braunschweiler, and R. Maia, "Speaker adaptation in DNN-based speech synthesis using d-vectors." in *INTERSPEECH 2017*, 2017, pp. 3404–3408.

[17] Y. Jia, Y. Zhang, R. J. Weiss, Q. Wang, J. Shen, F. Ren, Z. Chen, P. Nguyen, R. Pang, I. Lopez-Moreno *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *NeurIPS*, 2018.

[18] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *INTERSPEECH 2015*, 2015.

[19] J. S. Chung, J. Huh, S. Mun, M. Lee, H. S. Heo, S. Choe, C. Ham, S. Jung, B.-J. Lee, and I. Han, "In defence of metric learning for speaker recognition," in *INTERSPEECH 2020*, 2020, pp. 2977–2981.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017, pp. 5998–6008.

[21] W. Cai, J. Chen, and M. Li, "Exploring the encoding layer and loss function in end-to-end speaker and language recognition system," in *Proc. Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018, pp. 74–81.

[22] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE." *Journal of machine learning research*, vol. 9, no. 11, pp. 2579–2605, 2008.

[23] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "FastSpeech 2: Fast and high-quality end-to-end text to speech," *arXiv e-prints*, p. arXiv:2006.04558, 2021.

[24] H. Kawahara, I. Masuda-Katsuse, and A. De Cheveigne, "Restructuring speech representations using a pitch-adaptive time–frequency smoothing and an instantaneous-frequency-based f0 extraction: Possible role of a repetitive structure in sounds," *Speech communication*, vol. 27, no. 3-4, pp. 187–207, 1999.

[25] E. Cooper, C.-I. Lai, Y. Yasuda, and J. Yamagishi, "Can speaker augmentation improve multi-speaker end-to-end TTS?" in *INTERSPEECH 2020*, 2020, pp. 3979–3983.

[26] Z. Zeng, J. Wang, N. Cheng, T. Xia, and J. Xiao, "AlignTTS: Efficient feed-forward text-to-speech system without explicit alignment," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6714–6718.