



# TTS-by-TTS 2: Data-Selective Augmentation for Neural Speech Synthesis Using Ranking Support Vector Machine with Variational Autoencoder

Eunwoo Song<sup>1</sup>, Ryuichi Yamamoto<sup>2</sup>, Ohsung Kwon<sup>1</sup>, Chan-Ho Song<sup>1</sup>, Min-Jae Hwang<sup>1</sup>,  
Suhyeon Oh<sup>1</sup>, Hyun-Wook Yoon<sup>1</sup>, Jin-Seob Kim<sup>1</sup>, Jae-Min Kim<sup>1</sup>

<sup>1</sup>NAVER Corp., Seongnam, Korea

<sup>2</sup>LINE Corp., Tokyo, Japan

## Abstract

Recent advances in synthetic speech quality have enabled us to train text-to-speech (TTS) systems by using synthetic corpora. However, merely increasing the amount of synthetic data is not always advantageous for improving training efficiency. Our aim in this study is to selectively choose synthetic data that are beneficial to the training process. In the proposed method, we first adopt a variational autoencoder whose posterior distribution is utilized to extract latent features representing acoustic similarity between the recorded and synthetic corpora. By using those learned features, we then train a ranking support vector machine (RankSVM) that is well known for effectively ranking relative attributes among binary classes. By setting the recorded and synthetic ones as two opposite classes, RankSVM is used to determine how the synthesized speech is acoustically similar to the recorded data. Then, synthetic TTS data, whose distribution is close to the recorded data, are selected from large-scale synthetic corpora. By using these data for retraining the TTS model, the synthetic quality can be significantly improved. Objective and subjective evaluation results show the superiority of the proposed method over the conventional methods.

**Index Terms:** Speech synthesis, data augmentation, variational autoencoder, ranking support vector machine

## 1. Introduction

As the accuracy of acoustic modeling has increased following the revolution of deep neural networks, the synthetic quality of neural text-to-speech (TTS) systems has improved significantly [1–3]. However, these systems still have a major shortcoming in that a lot of training corpora are required to learn the complex nature of speech production [4].

To overcome this limitation, various studies employing data augmentation techniques have been proposed. For instance, Huybrechts et al. [5, 6] proposed using a well-trained voice conversion model to extend the speaking style of the target speaker’s TTS acoustic model; Wu et al. [7] proposed a speaker similarity-based data selection method from other speakers’ recordings for enlarging the training corpus of the TTS vocoding model. While those methods require external datasets to augment target speakers’ voices, Sharma et al. [8] proposed to generate a large-scale synthetic corpus within the same speaker’s model to distill the knowledge from the autoregressive (AR) WaveNet to the non-AR Parallel WaveNet.

Similar to Sharma’s work, our previous work proposed a TTS-by-TTS model in which a large-scale synthetic corpus generated by a well-designed TTS model is used to improve the quality of other TTS models [9]. One of the key ideas of this work was to collect a large number of text scripts while maintaining the recording scripts’ phoneme distribution. This enabled the model to simulate various phoneme combinations, resulting in significantly improving the TTS model’s stability with the unseen text.

However, increasing the amount of the corresponding synthetic waveform is not always advantageous for improving training efficiency [10], since this might cause negative effects if poorly generated waveforms are included. To address this problem, we propose a data-selective augmentation method for TTS systems. From a large-scale synthetic corpus, the proposed method can selectively choose the training data whose acoustic distribution is similar to the recordings. Specifically, we adopt a *variational autoencoder* (VAE) as a reference encoder for the TTS acoustic model [11]. VAEs are well known for capturing latent representations of feature distribution and have been employed as style encoders of controllable and/or expressive TTS tasks [12–15]. Similarly, our method utilizes a VAE to capture acoustic similarities between the acoustic features extracted from the natural recordings and synthesized by the TTS models.

We introduce a *ranking support vector machine* (RankSVM) [16], which is well known for scoring relative attributes between binary classes. Diverging from our previous work [10] that used OpenSMILE features [17], we employ latent feature vectors extracted from the learned VAE’s posterior distribution for recorded and synthetic ones. By setting them as two opposite classes, the RankSVM learns to determine *originalities* that represent how the distribution of the synthetic waveform is acoustically close to that of the natural recordings. On the basis of this originality, it is possible to selectively discard the synthetic waveforms whose attributes are dissimilar to the recordings; therefore, the modeling accuracy of the TTS system retrained with the remaining samples becomes significantly improved.

The objective and subjective evaluation results verified the superiority of the proposed method over the original system trained with recorded data alone and the similarly configured system retrained with all the synthetic data without any selection method. In particular, our method achieved 3.89 and 3.74 mean opinion score (MOS) for Tacotron 2 and FastSpeech 2 models, respectively, with 1,000 utterances of limited training data.

## 2. TTS-driven data augmentation

### 2.1. Database

The experiments used a phonetically balanced Korean corpus recorded by a Korean female professional speaker. The speech signals were sampled at 24 kHz with 16-bit quantization. In total, 1,000 (1.8 hours), 270 (0.5 hours), 130 (0.2 hours) utterances were used for training, validation, and testing, respectively.

### 2.2. Baseline TTS model

It is crucial to design a well-structured TTS system to synthesize high-quality speech database. Among various state-of-the-art models, we opted to pursue a VAE-Tacotron 2 model with a phoneme alignment approach (Figure 1a) thanks to its stable generation and competitive synthetic quality [18, 19]. Following the

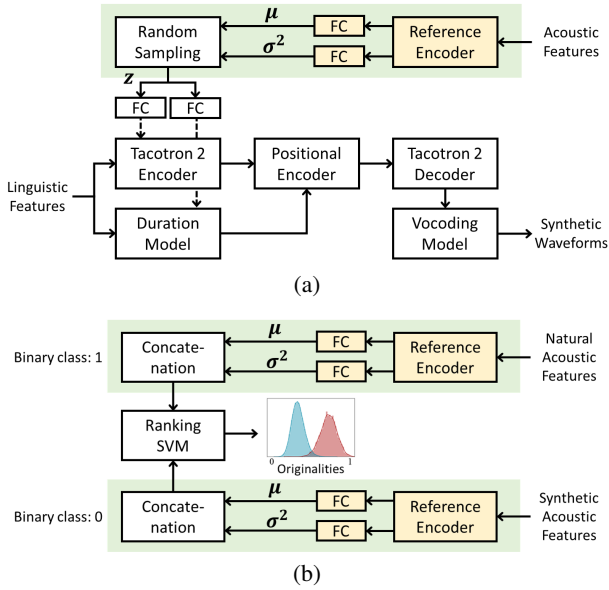


Figure 1: Block diagrams of (a) the baseline TTS model and (b) the RankSVM model. The symbols  $\mu$  and  $\sigma^2$  denote the VAE statistics (i.e., the mean and variance of the posterior distribution, respectively) to sample the latent variable  $z$ .

previous study in Zhang et al. [13], the prior and posterior distributions of the VAE are modeled by Gaussian distributions. Note that the use of VAE is beneficial not only for giving variations to the synthetic sample itself but also for capturing the acoustic characteristics of the natural recordings and the synthetic waveforms. Specifically, we leverage the statistics of the learned VAE’s posterior distribution (i.e., mean and variance) to measure acoustic similarities, which will be discussed in Section 3.

In the case of the vocoding models, we adopt a Parallel WaveGAN vocoder [20–22] based on a multi-band harmonic-plus-noise model (MB-HN-PWG [23]). Specifically, two separate WaveNet generators are trained to jointly learn the harmonic and noise characteristics of the speech, respectively. By employing the target speech’s periodicity in the multiple frequency bands, it is possible to produce speech outputs as naturally as recordings.

### 2.3. Large-scale data augmentation

To collect text scripts for synthesizing the large-scale speech waveforms, we crawled news articles from the NAVER website<sup>1</sup> [9, 10] and prepared 80,000 text scripts that were 80 times larger than the training utterances. Using these text scripts, the pre-trained TTS model described in the previous section generates<sup>2</sup> the corresponding 80,000 speech waveforms.

## 3. RankSVM-based data selection

Increasing the number of text scripts is beneficial for the target TTS model to learn the distribution of various phoneme combinations, enabling the model to generate more stable speech from the unseen text input [9]. However, we must carefully increase the number of corresponding synthetic waveforms to exclude poorly synthesized samples before retraining the target TTS model [10].

<sup>1</sup><https://news.naver.com>

<sup>2</sup>In the generation step, we condition the TTS acoustic model on VAE by using the centroid of latent vectors computed over all the training data [24].

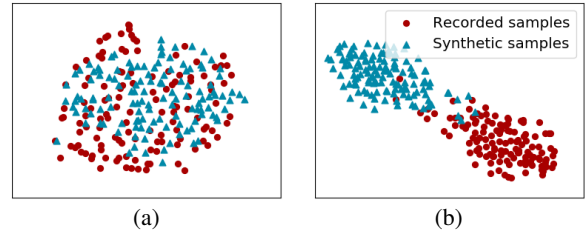


Figure 2: Distribution of the VAE’s posterior mean vector extracted from recorded (red dot) and synthetic (blue triangle) samples: (a) before and (b) after applying the VAE fine-tuning process.

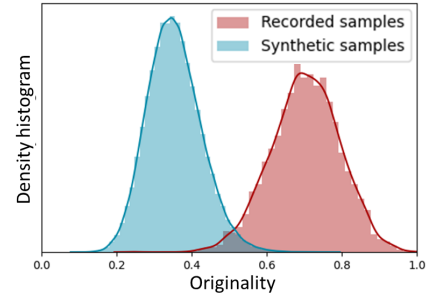


Figure 3: Density histograms of originality determined using RankSVM from the recorded (red boxes) and synthetic (blue boxes) waveforms, respectively.

To achieve this, we propose a RankSVM-based data selection method composed of three sub-steps: VAE fine-tuning, RankSVM training, and data selection.

### 3.1. VAE fine-tuning

To effectively train the RankSVM, it is very important to extract VAE statistics that adequately represent the distributions between the recorded and synthetic waveforms. However, the pre-trained TTS model is learned through the recorded database only, and it is difficult for the VAE model to capture the synthetic database’s distribution.

To address this problem, we fine-tune the pre-trained TTS model by using the TTS corpus combined with both recorded and synthetic corpora. Because both recorded and synthetic acoustic features are now included for training, the fine-tuned VAE model becomes aware of their acoustic distribution<sup>3</sup>. Figure 2 describes the VAE’s statistics approximated in the t-distributed stochastic neighbor embedding (t-SNE) space extracted from the test set [25]. It shows that applying the fine-tuning process (Figure 2b) represents the feature distributions better, which is expected to be beneficial for training the RankSVM model.

### 3.2. RankSVM

RankSVM is well known for learning relative attributes from two different classes [16]. As shown in Figure 1b, the proposed method sets the VAE feature vectors obtained from recorded and synthetic ones as two opposite classes, respectively. Then, a ranking function is learned to determine the *originality*, a real-valued score defined as how the distribution of the synthetic waveforms is acoustically similar to that of the natural recordings.

<sup>3</sup>The fine-tuned VAE is only used as a feature extractor of the RankSVM model. The large-scale data augmentation described in Section 2 is performed by the pre-trained model without the fine-tuning process.

Table 1: Distortions with a 95% confidence interval measured from two different groups: synthetic samples that have high 10% and low 10% originalities, respectively.

Range of originality	F0 RMSE (Hz)	LSD (dB)
High 10%	26.66±5.18	3.87±0.18
Low 10%	32.28±5.96	4.01±0.16

### 3.2.1. Ranking function

Assuming we have a training set  $T = \{t\}$  represented in  $\mathbb{R}^n$  by VAE features  $\{\mathbf{x}_t\}$ , and  $T = N \cup M$ , where  $N$  and  $M$  denote the recorded and synthetic set, respectively. The goal of RankSVM is to learn a ranking function defined as follows [26]:

$$r(\mathbf{x}_t) = \mathbf{w}^T \mathbf{x}_t, \quad (1)$$

with the following constraints:

$$\forall (i, j) \in O : \mathbf{w}^T \mathbf{x}_i > \mathbf{w}^T \mathbf{x}_j, \quad (2)$$

$$\forall (i, j) \in S : \mathbf{w}^T \mathbf{x}_i = \mathbf{w}^T \mathbf{x}_j, \quad (3)$$

where  $\mathbf{w}$ ,  $O$ , and  $S$  denote the weight vector of the ranking function, ordered set, and similar paired set, respectively. Specifically, the ordered set represents pairs of recorded and synthetic samples, and the similar set represents either pairs of synthetic samples or pairs of recorded samples.

Parikh et al. [16] proposed to solve the problem using Newton’s method [27]. However, this is not practical for large-scale data augmentation because of its quadratic computational complexity with regard to the number of training samples. Therefore, we adapted a stochastic gradient descent to accelerate the training [28]. Once we obtain an optimal  $\mathbf{w}$ , the originality of recorded and synthetic samples is calculated by Equation (1). Note that the originality scores are normalized to  $[0, 1]$  for convenience.

### 3.2.2. Relationship between originality and acoustic similarity

To further verify the relationship between the synthetic sample’s originality and its acoustic similarity with the natural recordings, we analyzed distortions in the generated acoustic features. In detail, we collected two different synthetic groups from the test set. One group consisted of synthetic samples that had high 10% originalities, whereas the other group consisted of synthetic samples that had low 10% originalities. Then, objective metrics, such as F0 root mean square error (RMSE) and log spectral distance (LSD), were measured from each group.

Table 1 shows the test results. It can be seen that the first group has much smaller generation distortions, which confirms that the synthetic waveforms with high originality have acoustically close characteristics to the natural recordings.

### 3.3. Data selection

On the basis of the synthetic waveform’s originality, we selectively discard the samples (e.g., those with low originality) that have dissimilar characteristics with the recordings. The entire acoustic model described in Section 2.2 is then retrained by using the remaining samples (e.g., those with high originality) together with the original recordings. Note that the proposed task is not applied to the vocoding model since it has been reported that using large sizes of training data is not important for training the vocoding model [29]. Consistently, our previous work also confirmed that including the augmented database to the training process was not that effective to improve the vocoding quality [9].

## 4. Experiments

### 4.1. TTS model details

#### 4.1.1. Feature extraction

The linguistic features consisted of 354-dimensional feature vectors containing 330 binary features for categorical contexts and 24 additional features for numerical contexts. The corresponding acoustic features were extracted using the improved time-frequency trajectory excitation vocoder at every 5 ms frame intervals [30]. Note that the feature dimension was 79 containing 40-dimensional line spectral frequencies, F0, gain, binary voicing flag (v/uv), a 32-dimensional slowly evolving waveform, and a 4-dimensional rapidly evolving waveform.

#### 4.1.2. Acoustic model

We adopted a VAE-Tacotron 2 model with a phoneme alignment approach [13, 18]. As shown in Figure 1a, the reference acoustic features were first fed into the VAE reference encoder composed of six convolutional layers followed by a gated recurrent unit layer [31]. Its output vector was then passed through two separate fully connected (FC) layers to generate the VAE mean and variance, respectively. Finally, the latent variables<sup>4</sup> were sampled via the reparameterization method [11].

Having input linguistic features, the duration model predicted phoneme-level duration through three FC layers followed by a unidirectional long short-term memory (LSTM) network. The encoder transformed the same linguistic features into high-level context vectors and these were upsampled to the frame resolution. Note that the encoder consisted of three convolution layers, a single bi-directional LSTM network, and a single FC layer.

The decoder was used to generate the output acoustic features. First, the previously generated acoustic features were fed into the PreNet [2], and those features and the context vectors from the encoder were then passed through two unidirectional LSTM layers, followed by two projection layers. Finally, residual elements of the generated acoustic features were passed through the PostNet [2] to improve generation accuracy.

Both the input linguistic and output acoustic features were normalized to have zero mean and unit variance. Xavier initializer [32] and RAdam optimizer [33] were used for initializing and updating the weights, respectively. More setup details are given in our previous work [9].

#### 4.1.3. Vocoding Model

The setups for training the MB-HN-PWG model followed our previous work in [23]. The harmonic WaveNet consisted of 20 dilated residual blocks with two exponentially increasing dilation cycles. On the other hand, the noise WaveNet consisted of 10 residual blocks with one exponentially increasing dilation cycle. In each WaveNet, 16 band-pass filters parameterized by windowed sinc functions with 255 filter taps were applied to divide the frequency bands. The model was trained for 400K steps with RAdam optimizer [33]. The discriminator weights were fixed for the first 100K steps, and both the generator and the discriminator were jointly trained for the rest 300K steps [23].

### 4.2. Objective and subjective evaluations

To evaluate the performance of the proposed system, we measured the distortions between the acoustic features obtained from the original recording and generated by the TTS models. The metrics

<sup>4</sup>They were simply added to the final layers of the duration model and the Tacotron 2 encoder, respectively. Note that two separated FC layers were used for projecting latent variables to each model.

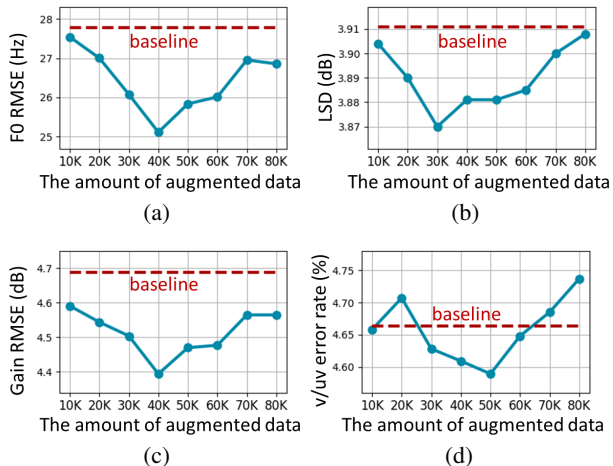


Figure 4: Objective evaluation results with respect to various amounts of augmented data used in the retraining process: the dashed red line represents the results of the baseline model learned with recorded data alone.

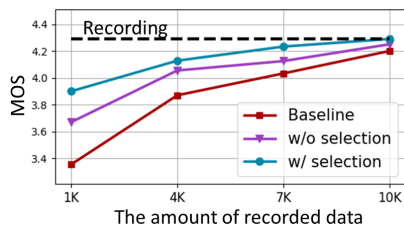


Figure 5: The MOS test results with respect to various amounts of recorded data: baseline model trained with recorded data alone (red square) and augmented models trained without (purple triangle) and with (blue dot) the proposed data selection method.

for measuring distortion were F0 RMSE, LSD, gain RMSE, and v/uv error rate.

Figure 4 shows the objective evaluation results with respect to various amounts of augmented data sets used in the retraining process (10K, 20K, ..., 80K utterances). The findings can be analyzed as follows: (1) The distortions decreased when the number of augmentation databases increased to some amount (about 40K utterances) because the modeling accuracy was also improved, depending on the amount of training databases. (2) However, distortions increased when the model was trained with more than 50K augmented utterances since the amount of poorly generated data increased as well. This implies the importance of data-selective augmentation in the TTS task, which will be further analyzed in the following subjective evaluations.

To evaluate the perceptual quality of the proposed system, we performed five-scale naturalness MOS listening tests<sup>5</sup>. In the test, twenty native Korean listeners were asked to make quality judgments about 20 randomly selected samples from the test set.

For comparison, we included the FastSpeech (FS) 2-based TTS system [34]. The structure was similar to the Tacotron 2-based baseline TTS system, but only the acoustic model was replaced with the non-AR Transformer networks [35]. The detailed setup for conducting the FS 2 model followed those in our previous work [9].

<sup>5</sup>Generated audio samples are available at the following URL: <https://sewplay.github.io/demos/txt2/>

Table 2: The MOS test results with a 95% confidence interval: the system trained with the proposed method is shown in bold-face. Note that the number of recorded utterances was 1K, and that of augmented utterances was represented as  $M$ .

System	Model type	$M$	Data selection	MOS
Test 1	Tacotron 2	-	-	3.35±0.11
Test 2	Tacotron 2	80K	No	3.66±0.11
<b>Test 3</b>	<b>Tacotron 2</b>	<b>40K</b>	<b>Yes</b>	<b>3.89±0.09</b>
Test 4	FastSpeech 2	-	-	3.30±0.12
Test 5	FastSpeech 2	80K	No	3.61±0.09
<b>Test 6</b>	<b>FastSpeech 2</b>	<b>40K</b>	<b>Yes</b>	<b>3.74±0.08</b>
Recording	-	-	-	4.29±0.08

Table 2 shows the MOS evaluation results for the TTS systems with respect to the different training conditions, and the analysis can be summarized as follows: (1) In both the AR and non-AR models, the systems retrained with the synthetic corpus performed better than those trained with recorded data alone. It means that increasing the training data with the augmented waveforms was effective for enhancing the synthetic quality. (2) Among the systems with augmentations, the proposed systems with the data-selective augmentation method (Tests 3 and 6) performed better than the systems without any selection method (Tests 2 and 5, respectively). As poorly generated synthetic samples were discarded based on originality, the proposed TTS systems synthesized more natural speech. (3) Consequently, our system achieved 3.89 and 3.74 MOS for the Tacotron 2 and FS 2 models, respectively, when the recorded data for training the model were limited to 1,000 utterances.

### 4.3. Additional experiments with enough recordings

To further confirm the effectiveness of the proposed method under the condition of the enough recordings, we conducted additional MOS tests by changing the number of recorded datasets from 1K to 10K utterances. In each case, we trained the baseline VAE-Tacotron 2 model, augmented 80K utterances, selected 40K utterances by using the fine-tuned VAE with the RankSVM model, and retrained the target TTS model. As shown Figure 5, the perceptual quality was improved as the amount of recorded data increased. Although the gap between the proposed and conventional methods was reduced as the synthetic quality was also improved, the data-selective augmentation method was still effective, even when the size of the source database increased enough.

## 5. Conclusion

We proposed a TTS-driven data-selective augmentation technique. From the large-scale synthetic corpora, a RankSVM with VAE’s posterior distribution determined the originality that represents how the acoustic characteristics of the generated speech was similar to those of the natural recordings. By selectively including the synthetic data with the recorded one, the performance of the retrained TTS system has been improved significantly. Future works include to extend the proposed method to augment other speaking styles for the emotional and/or expressive TTS models.

## 6. Acknowledgment

This work was supported by Clova Voice, NAVER Corp., Seongnam, Korea.

## 7. References

- [1] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomvrgiannakis, R. Clark, and R. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech*, 2017, pp. 4006–4010.
- [2] 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. Agiomvrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning WaveNet on Mel spectrogram predictions," in *Proc. ICASSP*, 2018, pp. 4779–4783.
- [3] N. Li, S. Liu, Y. Liu, S. Zhao, M. Liu, and M. T. Zhou, "Neural speech synthesis with Transformer network," in *Proc. AAAI*, 2019, pp. 6706–6713.
- [4] Y.-A. Chung, Y. Wang, W.-N. Hsu, Y. Zhang, and R. Skerry-Ryan, "Semi-supervised training for improving data efficiency in end-to-end speech synthesis," in *Proc. ICASSP*, 2019, pp. 6940–6944.
- [5] G. Huybrechts, T. Merritt, G. Comini, B. Perz, R. Shah, and J. Lorenzo-Trueba, "Low-resource expressive text-to-speech using data augmentation," in *Proc. ICASSP*, 2021, pp. 6593–6597.
- [6] M. S. Ribeiro, J. Roth, G. Comini, G. Huybrechts, A. Gabrys, and J. Lorenzo-Trueba, "Cross-speaker style transfer for text-to-speech using data augmentation," in *Proc. ICASSP (in press)*, 2022.
- [7] Y.-C. Wu, C.-H. Hu, H.-S. Lee, Y.-H. Peng, W.-C. Huang, Y. Tsao, H.-M. Wang, and T. Toda, "Relational data selection for data augmentation of speaker-dependent multi-band MelGAN vocoder," in *Proc. Interspeech*, 2021, pp. 3630–3634.
- [8] M. Sharma, T. Kenter, and R. Clark, "StrawNet: Self-training WaveNet for TTS in low-data regimes," in *Proc. Interspeech*, 2020, pp. 3550–3554.
- [9] M.-J. Hwang, R. Yamamoto, E. Song, and J.-M. Kim, "TTS-by-TTS: TTS-driven data augmentation for fast and high-quality speech synthesis," in *Proc. ICASSP*, 2021, pp. 6598–6602.
- [10] S. Oh, O. Kwon, M.-J. Hwang, J.-M. Kim, and E. Song, "Effective data augmentation methods for neural text-to-speech systems," in *Proc. ICEIC*, 2022, pp. 1–4.
- [11] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *Proc. ICLR*, 2014.
- [12] K. Akuzawa, Y. Iwasawa, and Y. Matsuo, "Expressive speech synthesis via modeling expressions with variational autoencoder," in *Proc. Interspeech*, 2018, pp. 3067–3071.
- [13] Y.-J. Zhang, S. Pan, L. He, and Z.-H. Ling, "Learning latent representations for style control and transfer in end-to-end speech synthesis," in *Proc. ICASSP*, 2019, pp. 6945–6949.
- [14] G. Sun, Y. Zhang, R. J. Weiss, Y. Cao, H. Zen, and Y. Wu, "Fully-hierarchical fine-grained prosody modeling for interpretable speech synthesis," in *Proc. ICASSP*, 2020, pp. 6264–6268.
- [15] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *Proc. PMLR*, 2021, pp. 5530–5540.
- [16] D. Parikh and K. Grauman, "Relative attributes," in *Proc. ICCV*, 2011, pp. 503–510.
- [17] F. Eyben, M. Wöllmer, and B. Schuller, "OpenSMILE - The Munich versatile and fast open-source audio feature extractor," in *Proc. ACM*, 2010, pp. 1459–1462.
- [18] T. Okamoto, T. Toda, Y. Shiga, and H. Kawai, "Tacotron-based acoustic model using phoneme alignment for practical neural text-to-speech systems," in *Proc. ASRU*, 2019, pp. 214–221.
- [19] E. Song, M.-J. Hwang, R. Yamamoto, J.-S. Kim, O. Kwon, and J.-M. Kim, "Neural text-to-speech with a modeling-by-generation excitation vocoder," in *Proc. Interspeech*, 2020, pp. 3570–3574.
- [20] R. Yamamoto, E. Song, and J. Kim, "Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *Proc. ICASSP*, 2020, pp. 6199–6203.
- [21] E. Song, R. Yamamoto, M.-J. Hwang, J.-S. Kim, O. Kwon, and J.-M. Kim, "Improved Parallel WaveGAN vocoder with perceptually weighted spectrogram loss," in *Proc. SLT*, 2021, pp. 470–476.
- [22] R. Yamamoto, E. Song, M.-J. Hwang, and J.-M. Kim, "Parallel waveform synthesis based on generative adversarial networks with voicing-aware conditional discriminators," in *Proc. ICASSP*, 2021, pp. 6039–6043.
- [23] M.-J. Hwang, R. Yamamoto, E. Song, and J.-M. Kim, "High-fidelity Parallel WaveGAN with multi-band harmonic-plus-noise model," in *Proc. Interspeech*, 2021, pp. 2227–2231.
- [24] S.-Y. Um, S. Oh, K. Byun, I. Jang, C. Ahn, and H.-G. Kang, "Emotional speech synthesis with rich and granularized control," in *Proc. ICASSP*, 2020, pp. 7254–7258.
- [25] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.
- [26] W. Deng, J. Hu, and J. Guo, "Linear ranking analysis," in *Proc. CVPR*, 2014, pp. 3638–3645.
- [27] O. Chapelle, "Training a support vector machine in the primal," *Neural computation*, vol. 19, no. 5, pp. 1155–1178, 2007.
- [28] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, "Pegasos: Primal estimated sub-gradient solver for svm," *Mathematical programming*, vol. 127, no. 1, pp. 3–30, 2011.
- [29] K. Matsubara, T. Okamoto, R. Takashima, T. Takiguchi, T. Toda, Y. Shiga, and H. Kawai, "Investigation of training data size for real-time neural vocoders on CPUs," *Acoust. Sci. Tech.*, vol. 42, no. 1, pp. 65–68, 2021.
- [30] E. Song, F. K. Soong, and H.-G. Kang, "Effective spectral and excitation modeling techniques for LSTM-RNN-based speech synthesis systems," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 25, no. 11, pp. 2152–2161, 2017.
- [31] Y. Wang, D. Stanton, Y. Zhang, R.-S. Ryan, E. Battenberg, J. Shor, Y. Xiao, Y. Jia, F. Ren, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," in *Proc. ICML*, 2018, pp. 5180–5189.
- [32] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. AISTATS*, 2010, pp. 249–256.
- [33] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, "On the variance of the adaptive learning rate and beyond," in *Proc. ICLR*, 2020.
- [34] Y. Ren, C. Hu, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "FastSpeech 2: Fast and high-quality end-to-end text-to-speech," in *Proc. ICLR*, 2021.
- [35] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T. Liu, "FastSpeech: Fast, robust and controllable text to speech," in *Proc. NeurIPS*, 2019, pp. 3165–3174.