Hierarchical and Multi-Scale Variational Autoencoder for Diverse and Natural Non-Autoregressive Text-to-Speech

Jae-Sung Bae\textsuperscript{1}, Jinhyeok Yang\textsuperscript{2}, Tae-Jun Bak\textsuperscript{3}, Young-Sun Joo\textsuperscript{3}

\textsuperscript{1}Samsung Research, Republic of Korea
\textsuperscript{2}Supertone, Inc., Republic of Korea
\textsuperscript{3}AI Center, NCSOFT Corp., Republic of Korea

\texttt{js3.bae@samsung.com, yangyangii@supertone.ai, \{happyjun, ysjoo555\}@ncsoft.com}

\section*{Abstract}
This paper proposes a hierarchical and multi-scale variational autoencoder-based non-autoregressive text-to-speech model (HiMuV-TTS) to generate natural speech with diverse speaking styles. Recent advances in non-autoregressive TTS (NAR-TTS) models have significantly improved the inference speed and robustness of synthesized speech. However, the diversity of speaking styles and naturalness are needed to be improved. To solve this problem, we propose the HiMuV-TTS model that first determines the global-scale prosody and then determines the local-scale prosody via conditioning on the global-scale prosody and the learned text representation. In addition, we improve the quality of speech by adopting the adversarial training technique. Experimental results verify that the proposed HiMuV-TTS model can generate more diverse and natural speech as compared to TTS models with single-scale variational autoencoders, and can represent different prosody information in each scale.

\textbf{Index Terms}: non-autoregressive text-to-speech, hierarchical variational autoencoder, multi-scale prosody modeling

\section*{1. Introduction}
Text-to-speech (TTS) is a typical one-to-many (O2M) mapping problem because human speaks speech in a diverse way. Thanks to recent advances in deep learning techniques, the neural TTS models, especially autoregressive TTS (AR-TTS) models [1–3], address this problem via conditioning on the previously generated acoustic features and have dramatically improved the naturalness of synthesized speech. However, because of this autoregressive architecture, these models have slow inference speed and instability in the inference phase such as skipping and repeating. Recently, non-autoregressive TTS (NAR-TTS) models have significantly improved the inference speed and robustness of synthesized speech [4–11]. However, because the NAR-TTS models cannot use previously generated acoustic features as context, the O2M problem becomes severe, which results in a performance gap between the AR-TTS models, and a lack of diversity.

Prosody modeling is a crucial factor in alleviating the O2M problem and improving the performance of the NAR-TTS model. In [5,6], the authors propose the NAR-TTS models that model the variance information of speech with explicit prosody features such as pitch and energy. In the inference phase, these models can generate speech with various speaking styles by changing the predicted prosody feature values. However, because they are trained in a supervised manner utilizing handcrafted prosody features (e.g., duration, pitch, and energy) as labels, other prosody information that is intrinsically presented in speech (e.g., emotion) may not be modeled.

Other approaches to address the O2M problem of speech are by representing the variance information of speech into latent variables in an unsupervised manner [12–15]. These methods can generate speech with various speaking styles by modifying the latent variables. In [12] and [13], the authors represent prosody in a global-scale. That is, the utterance-level speaking style of a speech is modeled as a single latent vector. On the other hand, in [14] and [15], the prosody is modeled in a local-scale as a sequence of latent variables. These approaches model the locally varying prosody, such as accents or prosody that differs in phoneme-level depending on the nuance, which cannot be captured in the global-scale prosody modeling approaches. However, because of the absence of global-scale representation, the local-scale prosody models cannot generate a speech with a consistent speaking style. To compensate for these drawbacks, the multi-scale prosody representation method was proposed in [16]. This method represents the global and local scale prosody in separate latent spaces and controls the prosody in multi-scale by providing different reference speech for each scale, respectively. However, to generate a speech with natural prosody, a reference speech that contains a similar text content to an input text is necessary in the inference phase.

Recently, Variational autoencoders (VAEs) [17] have achieved a promising performance in the TTS domain [8, 10, 18–20]. Because VAEs are a class of probabilistic generative models, TTS models with VAE-based prosody module can generate speech in multiple ways for the same text via sampling. Therefore, the VAEs are suitable to handle the O2M problem of TTS. Furthermore, they have the advantage that reference speech is not required in the inference phase. In [10], the authors proposed the NAR-TTS model incorporated with two different kinds of VAE modules, i.e., global and local scale VAEs, respectively. However, in our preliminary experiments, the expressiveness of the single-scale VAEs was limited. TTS models incorporated with the hierarchical VAE architectures [18, 20] have also been widely studied to improve the expressiveness of VAE-based TTS models, however, they have been studied with the AR-TTS models.

To improve the naturalness and expressiveness in NAR-TTS models, several previous approaches adopted the hierarchical structure [9, 21, 22]. In our previous work [9], we showed that adopting a proper hierarchical structure is important for the NAR-TTS models. However, we have focused on improving the naturalness and intelligibility of the NAR-TTS models instead of prosody modeling. In [21] and [22], the authors proposed the NAR-TTS model that adopts the explicit prosody module with a hierarchical structure. VQ-VAE [23] was used in [21]...
to better model the prosody but they only focused on the local-scale prosody modeling. In [22], the authors build a hierarchical structure by using utterance-level prosody predictors in addition to phoneme-level prosody predictors. However, as in [5,6], only the pre-defined prosody features are modeled in a supervised manner.

Taking advantage of the upper mentioned prosody modeling approaches, we propose an NAR-TTS model with hierarchical and multi-scale VAE (HiMuV-TTS) to better address the 02M problem in the NAR-TTS models. We conjecture that the prosody of speech should be modeled in multiple scales with a hierarchical structure. Global-scale prosody such as overall pitch and speaking speed should be determined first, and then local-scale prosody such as intonation within a word, duration of a phoneme, and pauses between words should be modeled depending on it. Accordingly, the HiMuV-TTS model first extracts the global-scale prosody embedding from the mel-spectrogram. Then, it is replicated and concatenated with the learned text representation and aligned with the mel-spectrogram to form local-scale prosody embeddings. Finally, the prosody embeddings of both scales are concatenated and provided as a condition to the decoder. To further improve the speech quality, we adopted the adversarial training technique in [24]. In the experiments, we verified that the HiMuV-TTS model generates more diverse and natural speech compared to the single linear layer, and become the posterior mean $\mu^t$ and standard deviation (SD) $\sigma^t$ of the global-scale latent variables. The global-scale latent variables $z^0 \sim N(\mu^t, (\sigma^t)^2)$ are then sampled from the normal distribution. $z^0$ is then replicated and concatenated with the hidden representation $H$ and forms $H^t$ through the hidden-encoder. Next, the multi-head attention layer learns the correlation between $H^t$ and $H^l$ and outputs phoneme-level features. The two layers of Bi-GRU followed by a single linear layer predict $\mu_n$ and $\sigma_n$, which are the posterior mean and SD of the local-scale latent variables of the $n$-th time-step, respectively. The local-scale latent variables $z_n^t \sim N(\mu_n^t, (\sigma_n^t)^2)$ is sampled from them, where $n = 1, 2, ..., N$. Finally, the replicated $z^0$ and $Z^t$ are concatenated and become the final prosody embedding $Z$. The prior over latent variables of both scales are assumed to follow a standard normal distribution. The model is optimized with the evidence lower bound (ELBO):

$$ ELBO = E_{q(Z|X,Y)}[\log p(X|Y,Z)] + L_{KL} $$

$$ L_{KL} = -\beta_1 \sum_{n=1}^{N} [D_{KL}(q(z_n^t|X, z^0, H)||p(z_n^t))] \\
- \beta_2 D_{KL}(q(z^0|X)||p(z^0)) $$

2.2. HiMuV Encoder

The HiMuV-encoder first represents the global-scale prosody from the mel-spectrogram as global-scale latent variables $z^0$ and then models the local-scale latent variables $Z^t = \{z^t\}_{n=1}^{N}$ conditioned on $z^0$ and the hidden representation $H$. The architecture of the HiMuV-encoder is depicted in Figure 1b. First, to model the global-scale prosody, it encodes the mel-spectrogram $X$ into the intermediate feature $H^0$ with the mel-encoder. After two layers of a bidirectional GRU [25] (Bi-GRU), it is averaged through the time axis, projected through a single linear layer, and become the posterior mean $\mu^t$ and standard deviation (SD) $\sigma^t$ of the global-scale latent variables. The global-scale latent variables $z^0 \sim N(\mu^t, (\sigma^t)^2)$ are then sampled from the normal distribution. $z^0$ is then replicated and concatenated with the hidden representation $H$ and forms $H^t$ through the hidden-encoder. Next, the multi-head attention layer learns the correlation between $H^t$ and $H^l$ and outputs phoneme-level features. The two layers of Bi-GRU followed by a single linear layer predict $\mu_n$ and $\sigma_n$, which are the posterior mean and SD of the local-scale latent variables of the $n$-th time-step, respectively. The local-scale latent variables $z_n^t \sim N(\mu_n^t, (\sigma_n^t)^2)$ is sampled from them, where $n = 1, 2, ..., N$. Finally, the replicated $z^0$ and $Z^t$ are concatenated and become the final prosody embedding $Z$. The prior over latent variables of both scales are assumed to follow a standard normal distribution. The model is optimized with the evidence lower bound (ELBO):
2.4. Final Loss Function

Equation (8) is the final loss function used for training, where $\gamma$ and $\delta$ are the weights of the posterior mean prediction loss and feature matching loss, respectively.

$$L_{\text{final}} = L_{\text{recon}} + L_{\text{KL}} + \gamma L_{\text{post}} + L_{\text{adv}} + \delta L_{\text{fm}} \quad (8)$$

3. Experiments

3.1. Experimental Setup

The LJ speech dataset [27] was used for the experiments. It consists of 13,100 utterances from a single English female speaker with a total of approximately 24 hours. It was down-sampled to 22.05KHz. We split 131 utterances for testing and all the remaining utterances were used for training. An 80-bin mel-spectrogram with a hop size of 256 and a window size of 1024 was computed. The phoneme-level duration values were carried out using the Montreal forced alignment [28] tool. The phoneme-level pitch values were obtained by averaging the pitch values for each phoneme. The PRAAT toolkit [29] was used to extract pitch values.

The text-encoder and decoder of the HiMuV-TTS model have the same architecture as described in [5]. For the mel-encoder and the hidden-encoder of the HiMuV-encoder, the architecture in [30] was adopted. The hidden dimensions of the Bi-GRU and attention layer of the HiMuV-encoder were both 128. The attention layer had two heads. The dimensions of $z^g$ and $z^l$ were 32 and 16, respectively. The architecture of the duration and pitch predictors consists of two layers of LSTM [31] with 256 hidden dimensions followed by a single projection layer. The one dimensional convolutional layer with the kernel size of 3 and the stride of 1 was used for pitch embedding.

For the training, the Adam optimizer [32] was used to optimize the network with an initial learning rate of 0.002 and a batch size of 32. It was trained for 200K iterations. To avoid the collapse in VAEs, the KL-weight schedule was applied for $\beta_g$ and $\beta_l$. Each was linearly increased from 0 to $10^{-4}$ and $10^{-6}$, respectively, from 10K to 60K. For the loss weights, $\alpha$, $\gamma$, and $\delta = 0.01$ were used. As a neural vocoder, Vocoder [33], trained using the same speech dataset, was used.

Four models were used for the comparison: the FastPitch [5] model, the GANSpeech model [24], and the global-scale VAE (GVAE) and the local-scale VAE (LVAE) based TTS models. The GVAE and LVAE models were chosen as single-scale VAE models. The VAE modules of the GVAE and LVAE models had the similar architecture as those in [10], but the mel-encoder layer of the HiMuV-encoder was adopted instead of LConv layers. For the fair comparison, we changed the baseline TTS model of all comparison models to the FastPitch model.

3.2. Sample Diversity Improvement

To verify the sample diversity improvement of the HiMuV-TTS model, we synthesized 100 different speech samples1 from the GVAE, LVAE, and HiMuV-TTS models for each sentence in the test dataset. The SD of the utterance length ($\sigma_l$), average energy ($\sigma_e$), average pitch ($\sigma_p$), and average pitch SD within utterance ($\sigma_{p_e}$) of the LVAE model results in unnatural speech (Table 2).

| Model        | $\sigma_l$ (s) | $\sigma_e$ (dB) | $\sigma_p$ (Hz) | $\sigma_{p_e}$ (Hz) |
|--------------|----------------|-----------------|-----------------|---------------------|
| GVAE         | 0.31           | 0.75            | 10.54           | 8.78                |
| LVAE         | 0.13           | 0.84            | 21.48           | 9.41                |
| HiMuV-TTS    | **0.50**       | **1.02**        | **12.01**       | **10.44**           |
| HiMuV-TTS-G  | 0.33           | 0.86            | 11.82           | 10.39               |
| HiMuV-TTS-L  | 0.36           | 0.71            | 2.46            | 5.37                |

1Audio samples can be found online: https://nc-ai.github.io/speech/publications/himuvtts

2The text was “They entered a stone cold room and were presently joined by the prisoner.”
tent pitch contour. Meanwhile, the HiMuV-TTS model could achieve high SD values generally for overall prosody features because it could well represent the various prosody attributes in a hierarchical and multi-scale way.

3.3. Latent Representations of Each Scale

Next, we compared the HiMuV-TTS models sampled in \( \tau \) but with fixed \( Z^g \) (HiMuV-TTS-G) and sampled in \( Z^l \) but with fixed \( z^g \) (HiMuV-TTS-L). The results are displayed in Table 1 and Figure 2. Unlike the GVAE and the LVAE model, the HiMuV-TTS model modeled the utterance-level prosody attributes such as overall tone or speaking style into \( z^g \), and the remaining locally varying prosody was modeled in \( Z^l \). Therefore, the HiMuV-TTS-G model had high \( \sigma_g \) and \( \sigma_l \) values, while those of the HiMuV-TTS-L model were low. The utterance length was varied by both scales of latent variables, but was changed for different reasons. As illustrated in Figure 3, the overall speaking speed of generated speech differed by sampling in the global-scale. Meanwhile, sampling in the local-scale varied the short pauses while the overall speaking speed was preserved.

3.4. Intelligibility

To evaluate the intelligibility of the synthesized speech, we computed the word error rate (WER) for each model. The conformer-based speech recognition model [34] trained with the LibriSpeech dataset [35] was used. The results are shown in Table 2. The GANSpeech model had a significantly lower WER value compared to the FastPitch model. This was because the adversarial training technique was adopted in the GANSpeech model. The WER results of the HiMuV-TTS model were slightly higher than the GANSpeech model, because of the diverse speaking style generated by the HiMuV modules. Meanwhile, the HiMuV-TTS model with the temperature value \( \tau = 0 \), which was multiplied to the prior SD, achieved the lowest WER results. When we analyzed the generated speech sample of the GVAE model, we observed that the speaking speed was commonly lower than the other models as shown in Figure 2. Therefore, the GVAE model had the second-lowest WER results.

3.5. Subjective Evaluation

To evaluate the naturalness of the synthesized speech, we conducted the mean opinion score (MOS) test. Seventeen samples were generated from each model. A total of 17 native English speakers participated and scored the naturalness of the generated speech samples from 1 to 5. The results are summarized in Table 2. The HiMuV-TTS model achieved similar MOS results as the FastPitch and GANSpeech models while generating speech with diver speaking styles. Furthermore, it achieved the best scores with \( \tau = 0 \). The LVAE model had significantly low MOS results because of the inconsistent prosody of generated speech among the text sequences.

4. Conclusion

We propose a hierarchical and multi-scale VAE architecture to generate diverse and natural speech for the NAR-TTS model. The HiMuV-TTS model first determines the global-scale prosody. The local-scale prosody of speech is then determined via conditioning on the global-scale prosody and a learned text representation. Additionally, we adopt the adversarial training to improve speech quality. Through experiments, we showed that the proposed HiMuV-TTS model can generate speech with more naturalness and diversity as compared with the TTS models with single-scale VAEs.
5. References

[1] J. Shen, R. Pang, R. Weiss, M. Schuster, N. Jaitly, Z. Yang et al., “Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions,” in Proc. ICASSP, 2018, pp. 4779–4783.

[2] W. Ping, K. Peng, A. Gibiansky, S. Arik, A. Kannan, S. Narang et al., “Deep voice 3: 2000-speaker neural text-to-speech,” in Proc. Int. Conf. on Learning Representations, 2018.

[3] J. S. Bae, H. Bae, Y. S. Joo, J. Lee, G. H. Lee, and H. Y. Cho, “Speaking speed control of end-to-end speech synthesis using sentence-level conditioning,” in Proc. Interspeech, 2020, pp. 4402–4406.

[4] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao et al., “Fastspeech: Fast, robust and controllable text to speech,” in Proc. Advances in Neural Information Processing Systems, 2019, pp. 3165–3174.

[5] A. Łańcucki, “Fastpitch: Parallel text-to-speech with pitch prediction,” in Proc. ICASSP, 2021, pp. 6588–6592.

[6] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao et al., “FastSpeech 2: Fast and high-quality end-to-end text to speech,” in Proc. Int. Conf. on Learning Representations (ICLR), 2021.

[7] J. Kim, S. Kim, J. Kong, and S. Yoon, “Glow-TTS: A generative flow for text-to-speech via monotonot alignment search,” in Proc. Advances in Neural Information Processing Systems, 2020, pp. 8067–8077. [Online]. Available: https://proceedings.neurips.cc/paper/2020/file/5c3b99e8f92532e5ad1556e3ceca0tc-Paper.pdf

[8] J. Kim, J. Kong, and J. Son, “Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech,” in Proc. Int. Conf. on Machine Learning, 2021, pp. 5530–5540.

[9] J. S. Bae, T. Bak, Y. S. Joo, and H. Y. Cho, “Hierarchical context-aware transformers for non-autoregressive text to speech,” in Proc. Interspeech, 2021, pp. 3610–3614.

[10] I. Elias, H. Zen, J. Shen, Y. Zhang, Y. Jia, R. J. Weiss et al., “Parallel Tacotron: Non-autoregressive and controllable tts,” in Proc. ICASSP, 2021, pp. 5709–5713.

[11] T. Bak, J.-S. Bae, H. Bae, Y.-I. Kim, and H.-Y. Cho, “FastPitch-Formant: Source-filter based decomposed modeling for speech synthesis,” in Proc. Interspeech, 2021, pp. 116–120.

[12] Y. Wang, D. Stanton, Y. Zhang, R. Skerritt-Ryan, E. Battenberg, J. Shor et al., “Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis,” in Proc. Int. Conf. on Machine Learning (ICML), 2018, pp. 5186–5190.

[13] R. Skerritt-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor et al., “Towards end-to-end prosody transfer for expressive speech synthesis with tacotron,” in Proc. Int. Conf. on Machine Learning, 2018, pp. 4693–4702.

[14] Y. Lee and T. Kim, “Robust and fine-grained prosody control of end-to-end speech synthesis,” in Proc. ICASSP, 2019, pp. 5911–5915.

[15] V. Klimkov, S. Ronanaki, J. Rohanke, and T. Drugman, “Fine-grained robust prosody transfer for single-speaker neural text-to-speech,” in Proc. Interspeech, 2019, pp. 4440–4444.

[16] X. Li, C. Song, J. Li, Z. Wu, J. Jia, and H. Meng, “Towards multiscale style control for expressive speech synthesis,” in Proc. Interspeech, 2021, pp. 4673–4677.

[17] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” in Proc. Int. Conf. on Learning Representations, 2014.

[18] W.-N. Hsu, Y. Zhang, R. Weiss, H. Zen, Y. Wu, Y. Cao et al., “Hierarchical generative modeling for controllable speech synthesis,” in Proc. Int. Conf. on Learning Representations, 2019.

[19] Y. Lee, J. Shin, and K. Jung, “Bidirectional variational inference for non-autoregressive text-to-speech,” in Proc. Int. Conf. on Learning Representations, 2021.

[20] 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.

[21] C. M. Chien and H. Y. Lee, “Hierarchical prosody modeling for non-autoregressive speech synthesis,” in 2021 IEEE Spoken Language Technology Workshop (SLT), 2021, pp. 446–453.

[22] T. Raitio, J. Li, and S. Seshadri, “Hierarchical prosody modeling and control in non-autoregressive parallel neural tts,” in Proc. ICASSP, 2022, pp. 7587–7591.

[23] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, “Neural discrete representation learning,” in Proc. Advances in Neural Information Processing Systems, vol. 30, 2017.

[24] J. Yang, J.-S. Bae, T. Bak, Y.-I. Kim, and H.-Y. Cho, “GANSpeech: Adversarial training for high-fidelity multi-speaker speech synthesis,” in Proc. Interspeech 2021, 2021, pp. 2202–2206.

[25] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” in NIPS 2014 Workshop on Deep Learning, 2014.

[26] G. Sun, Y. Zhang, R. J. Weiss, Y. Cao, H. Zen, A. Rosenberg et al., “Generating diverse and natural text-to-speech samples using a quantized fine-grained vae and autoregressive prosody prior,” in Proc. ICASSP, 2020, pp. 6699–6703.

[27] K. Ito and L. Johnson, “The LJ speech dataset,” https://keithito.com/LJ-Speech-Dataset/, 2017.

[28] M. McAuliffe, M. Socolof, S. Mihac, M. Wagner, and M. Sondergerg, “Montreal Forced Aligner: Trainable text-speech alignment using kaldi,” in Proc. Interspeech, 2017, pp. 498–502.

[29] P. Boersma, “Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound,” in Proc. institute of phonetic sciences, 1993, pp. 97–110.

[30] D. Min, D. B. Lee, E. Yang, and S. J. Hwang, “Meta-stylespeech: Multi-speaker adaptive text-to-speech generation,” in Proc. Int. Conf. on Machine Learning, 2021, pp. 7748–7759.

[31] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, no. 8, pp. 1735–1780, 1997.

[32] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv:1412.6980, 2014.

[33] J. Yang, J. Lee, Y. Kim, H.-Y. Cho, and I. Kim, “VocGAN: A high-fidelity real-time vocoder with a hierarchically-nested adversarial network,” Proc. Interspeech, 2020, pp. 200–204, 2020.

[34] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu et al., “Conformer: Convolution-augmented transformer for speech recognition,” in Proc. Interspeech, 2020, pp. 5036–5040.

[35] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: An asr corpus based on public domain audio books,” in Proc. ICASSP, 2015, pp. 5206–5210.