INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Speech Synthesis with Self-Supervisedly Learnt Prosodic Representations

Zhao-Ci Liu[1], *Zhen-Hua Ling*[1⋆], *Ya-Jun Hu*[2], *Jia Pan*[2], *Yun-Di Wu*[2], *Jin-Wei Wang*[2]

[1]National Engineering Research Center of Speech and Language Information Processing, University of Science and Technology of China, Hefei, P.R.China
[2]iFLYTEK Research, iFLYTEK Co. Ltd., China

zcliu8@mail.ustc.edu.cn, zhling@ustc.edu.cn, {yjhu, jiapan, ydwu2, jwwang15}@iflytek.com

Abstract

This paper presents S4LPR, a Speech Synthesis model conditioned on Self-Supervisedly Learnt Prosodic Representations. Instead of using raw acoustic features, such as F0 and energy, as intermediate prosodic variables, three self-supervised speech models are designed for comparison and are pre-trained on large-scale unlabeled data to extract frame-level prosodic representations. In addition to vanilla wav2vec 2.0, the other two pre-trained models learn representations from LPC residuals or adopt a multi-task learning strategy to focus on the prosodic information in speech. Based on FastSpeech2 and PnGBERT, our acoustic model is built with the learned prosodic representations as intermediate variables. Experimental results demonstrate that the naturalness of speech synthesized using S4LPR is significantly better than the FastSpeech2 baseline.

Index Terms: speech synthesis, self-supervised learning, LPC analysis, multi-task learning

1. Introduction

Speech synthesis, also known as text-to-speech (TTS), has already achieved a high degree of naturalness when producing broadcasting-style speech using modern neural network-based methods [1–4]. However, in scenarios requiring rich expressiveness, there is still a clear gap between the naturalness of synthetic speech and human recordings, which is mainly reflected in prosody. Prosody describes the subjective perception of intonation, pauses, stress, etc., and is related with the variations in fundamental frequency (F0), syllable duration, and energy. Depending on the speaker's intention or contextual information, the same text may be uttered with different prosodic characteristics, which is known as the "one-to-many" mapping issue [5] and increases the difficulty of synthesizing expressive speech.

Plenty of studies have attempted to address the one-to-many mapping issue by using intermediate prosodic representations between linguistic features and acoustic features to facilitate acoustic modeling. Some of them, such as FastSpeech2 [2] and FastPitch [6] adopted prosody-related acoustic features, e.g., F0 and energy, as prosodic representations. However, the accurate extraction of F0 is still challenging. Furthermore, F0 is a low-level descriptor with entangled stress, intonation and speaker information. The difficulty of F0 prediction usually leads to over-smoothed F0 contours. FastSpeech2 attempted to solve this problem by applying continuous wavelet transform (CWT) [7] on F0 and predicting the F0 spectrogram instead. Some other studies designed auto-encoder networks to learn intermediate prosodic representations from raw spectral features [8–11]. In these methods, the encoder for extracting prosodic representation and the decoder for speech generation were jointly trained under supervised criteria, e.g., spectrum reconstruction. However, the prosodic representation extractor in these methods need to be jointly trained with the generation model, thus its generalization ability is limited and cannot benefit from large-scale unlabeled data.

On the other hand, self-supervised speech models, such as wav2vec 2.0 [12] and data2vec [13], have been proposed in recent years. These models adopted self-supervision techniques, such as contrastive learning or mask prediction, to learn speech representations from large-scale training sets, and the learned representations benefitted several downstream tasks, including automatic speech recognition (ASR), speaker diarization, emotion recognition, etc. [14]. For speech synthesis, Lim et al. [15] and Siuzdak et al. [16] replaced the target Mel-spectrogram for acoustic modeling with the embeddings given by wav2vec 2.0. Regarding with prosody modeling, Polyak et al. [17] proposed a pre-trained method to extract low-bitrate representations for speech and resynthesis it, which is not a complete TTS model as it lacks text input. Chen et al. proposed speech BERT [18] to extract fine-grained segment-level prosodic representations. But this model relied on text input at the training stage and cannot utilize unlabeled speech data. Our concurrent work ProsodyBERT [19] proposed a similar idea with us, but employing different input and network structures.

Therefore, this paper proposes to build self-supervised speech models to derive prosodic representations for speech synthesis. Three representation extraction models are designed for comparison and are pre-trained on large-scale unlabeled data under self-supervised criteria. The first model is the vanilla wav2vec 2.0 with some modified hype-parameters to make it more compatible with prosody modeling. To build the second model, the raw waveforms for training wav2vec 2.0 are replaced by the residual waveforms of linear predictive coding (LPC) inverse filtering to remove the content information in the extracted representations. The third model is built to combine the advantages of the above two models by multi-task learning. It takes raw waveforms as input and predicts both waveform-encoded and residual-encoded targets. Then, based on FastSpeech2 and PnGBERT [20], an acoustic model for speech synthesis is designed using the extracted prosodic representations as intermediate variables. Experimental results on a Chinese audiobook dataset show that our proposed method outperformed the FastSpeech2 baseline, and the third model achieved the best objective performance on the test set.

2. Proposed Methods

2.1. Prosodic Representation Learning

2.1.1. Vanilla wav2vec 2.0

wav2vec 2.0 [12] learns high-level representations from waveforms without labels, and its structure consists of a convolu-

⋆ Corresponding author.

10.21437/Interspeech.2023-1292

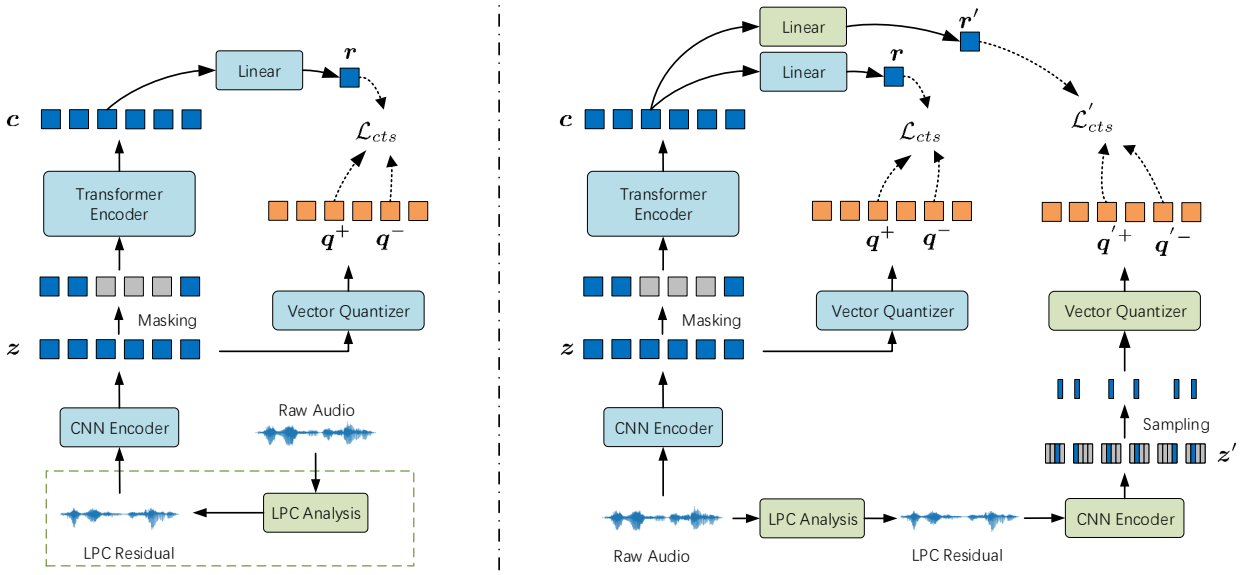

Figure 1: *The structure of our proposed representation extraction models. The left sub-figure shows the vanilla wav2vec 2.0 model (skipping the dashed box) or the wav2vec 2.0 model for LPC residuals (with the dashed box). The right sub-figure shows the wav2vec 2.0 model with multi-task training.*

tional encoder and a multi-layer Transformer, as shown in the left part of Figure 1. The input of the convolutional encoder is the raw waveform, and the output is the encoded feature $\boldsymbol{z}$. The Transformer encoder takes $\boldsymbol{z}$ as input and produces contextual representation $\boldsymbol{c}$. Masking is applied to the convolutional features at the Transformer input. The input data $\boldsymbol{z}$ is sliced into G equal parts, and all parts are passed to a vector quantizer. Following quantization, the output vectors are concatenated to obtain $\boldsymbol{q}$. This nonlinear mapping in quantizer is learnt via back-propagation and Gumbel-Softmax [21] with a codebook size of V. $\boldsymbol{r}$ is the output of $\boldsymbol{c}$ through a linear layer. wav2vec 2.0 use contrastive learning to minimize the distance between contextualized representations and quantized target vectors. Anchors are taken at masked timesteps only, the positive sample is chosen as the quantized vector at the same time step, and negatives are sampled from other masked timesteps. Prediction results are determined by the cosine similarities between $\boldsymbol{r}$ and $\boldsymbol{q}$. The loss function is the weighted sum of three items, contrastive loss $\mathcal{L}_{cts}$, diversity loss $\mathcal{L}_d$, and an L2 regularization term $\mathcal{L}_{pen}$ as

$$\mathcal{L} = \mathcal{L}_{cts} + \alpha\mathcal{L}_d + \beta\mathcal{L}_{pen}. \quad (1)$$

In this paper, for better capturing prosodic information, the respective field of the encoder is increased from 25ms to 50ms considering the supersegmental character of prosody. The hidden size of the model is reduced for faster training.

2.1.2. *wav2vec 2.0 for LPC residuals*

The representations from vanilla wav2vec 2.0 contain lots of content and speaker identity information. In order to remove such prosody-irrelevant information, we propose to replace the raw waveforms input with LPC residuals for model training. The LPC residuals [22, Chapter 9] are calculated as follows. Firstly, LPC coefficients are estimated using the Levison-Durbin algorithm. Then, the raw waveforms are inverse-filtered using the linear predictor to extract residual signals, which mainly describe the excitations in speech production. The calculation of LPC residual is more robust than F0 estimation. The spectra of LPC residuals are flat with the formant information removed. Using LPC residuals as the input of wav2vec 2.0, the encoded vectors contain less content information and retains mostly the F0 as well as energy information.

2.1.3. *wav2vec 2.0 with multi-task learning*

On the other hand, removing content information by using LPC residuals may increase the difficulty of contextual modeling in pre-training. Therefore, a multi-task learning strategy is proposed here to combine the advantages of the above two models. We add a residual encoder to vanilla wav2vec 2.0, turning it into a multi-task model with a secondary task. The structure of the model is shown in the right part of Figure 1. The proposed model is still trained in a fully self-supervised manner, except that the predicted targets are changed from one to two, and the loss function is the weighted sum of six terms, as

$$\mathcal{L} = \mathcal{L}_{cts} + \alpha\mathcal{L}_d + \beta\mathcal{L}_{pen} + \gamma(\mathcal{L}'_{cts} + \alpha\mathcal{L}'_d + \beta\mathcal{L}'_{pen}). \quad (2)$$

$\mathcal{L}'_{cts}$, $\mathcal{L}'_d$ and $\mathcal{L}'_{pen}$ are contrastive loss, diversity loss and L2 regularization term of the secondary task, respectively.

By constraining the linear prediction layer with a low dimensionality as a bottleneck, the obtained prosodic representation features are compressed to obtain $\boldsymbol{r}'$. In order to obtain a representation that represents the overall prosody variation of a piece of audio in addition to local details, the receptive fields of the residual encoder overlap with each other, and its hop size is n-times smaller than that of the waveform encoder. Thus, one frame of $\boldsymbol{q}$ corresponds to n frames of $\boldsymbol{q}'$. Then we randomly sample one frame of $\boldsymbol{q}'$ in every n frames so that $\boldsymbol{q}$ and $\boldsymbol{q}'$ are of the same length.

2.2. Synthesis model with learnt prosodic representations

An acoustic model is designed based on FastSpeech2 to integrate the self-supervisedly learnt prosodic representations for speech synthesis, as shown in Figure 2. The original configu-

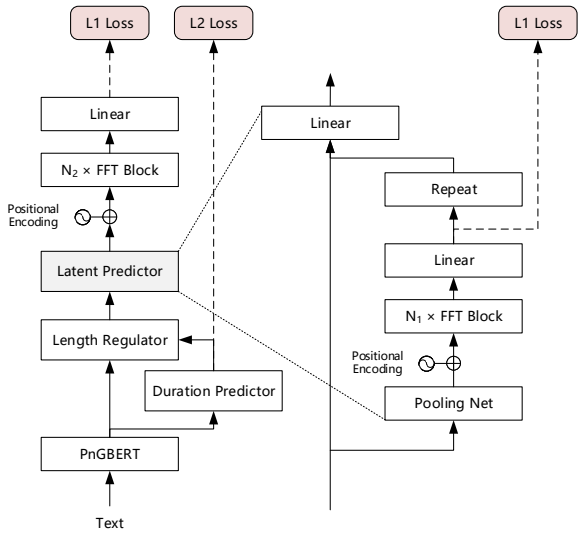

Figure 2: *The structure of our proposed synthesis model.*

rations of FastSpeech2 use phonemes as input and cannot distinguish homophones. In this paper, the text encoder in the model is replaced by a pre-trained PnGBERT [20], which combines phoneme and word inputs and provides semantic information. The original variance adapters for predicting energy and pitch are replaced by a Transformer-based latent predictor for predicting intermediate representations $\boldsymbol{r}$ or $\boldsymbol{r}'$. The input to the latent predictor is a frame-level text embedding that has been expanded using phoneme duration. The text sequences are downsampled by a pooling network which contains one or two layers of convolution and max-pooling. Then the sequences are passed through N_1 feed-forward Transformer (FFT) blocks [2] and a linear layer to predict latent representations. The output representations are repeated and concatenated with the text embedding. After dimension adjustment using a linear layer, they are used as the input of the Mel-spectrogram decoder.

The model is trained by optimizing the L2 loss between the predicted and ground-truth phoneme duration, and the L1 loss between the predicted and ground-truth Mel-spectrograms and latent representations.

3. Experiments

3.1. Datasets

In our experiments, speech synthesis models were trained on our internal dataset, which was derived from the audiobook speech of a Chinese male speaker. Only the narrative utterances were used in the experiment, which were about 25 hours. 14581 utterances were used for training, 168 utterances for validation, and 164 utterances for test. The prosodic representation extraction models were self-supervisedly trained using an internal dataset which contained approximately 519 hours of recordings for speech synthesis from 82 speakers. 280k sentences were used for training and 4094 sentences were used for validation. All audios were sampled at 16 kHz. The F0 data were extracted using STRAIGHT [23].

3.1.1. System Construction

The following five models were built for comparison.

FS2: We followed the configurations in the FastSpeech2 paper, except we enlarged the receptive field of the pitch predictor and energy predictor as [24] and replaced the text encoder with PnGBERT [20]. The PnGBERT used in our experiments was trained on a database with 130 million sentences of Chinese and English text. The inputs to the model had two segments, which were phoneme and character representations of the same sentence. The segment embedding, token position embedding, word position embedding and prosodic position embedding were added to the inputs. The main structure of the model was a 12-layer transformer with a hidden-size of 512, trained with the MLM criterion [25]. The phoneme part of the output sequence was taken out and transformed by a linear layer to obtain the 256-dimensional text embedding, which was used to replace the output of the text encoder in the original configures of FastSpeech2. In the synthesis model training, the first 9 layers of PnGBERT were fixed.

S4LPR_V: The model was built using the prosodic representations extracted by the vanilla wav2vec 2.0 introduced in Section 2.1.1. The wav2vec 2.0 model was trained based on the open-source implementation in fairseq [26]. The convolutional audio encoder contained 8 convolutions with 256 channels, kernel widths (10,3,3,3,3,3,2,2), and strides (5,2,2,2,2,2,2,2). Thus, the encoded output had a stride of 40ms between each sample and a receptive field of 50ms. The masking strategy was similar to [12]. We randomly sampled a certain proportion $p = 0.065$ of all time-steps to be starting indices and then masked the subsequent M=5 times-steps, i.e, 200ms of audio. The Transformer encoder contained 12 layers with 384 channels. We used $G = 2$ and $V = 320$ for the quantization module and $\alpha = 0.1$ and $\beta = 10$ for loss weights. The output $\boldsymbol{r}$ was used as the prosodic representation in speech synthesis, which had 256 channels. The latent decoder in the synthesis model had 6 FFT Blocks and the hidden size was 256.

S4LPR_LPC: The model was built using the prosodic representations extracted by the pre-trained model introduced in Section 2.1.2. The order of LPC analysis was set to 16. The LPC residuals were downsampled to 4 kHz. The convolutional LPC residual encoder contained 6 convolutions with 256 channels, kernel widths (10,3,3,3,2,2), and strides (5,2,2,2,2,2). We used the output $\boldsymbol{r}$ as prosodic representations in the synthesis model, which had 256 channels.

S4LPR_MT: The model was built using the prosodic representations extracted by the pre-trained model introduced in Section 2.1.3. The convolutional LPC residual encoder contained 6 convolutions with 128 channels, kernel widths (10,3,3,3,2,2), and strides (5,2,2,2,1,1). The last layer of the encoder was changed to dilation convolution with dilation=2. Thus, the encoded output had a stride of 10ms between each sample and a receptive field of 50ms. After that, one frame out of every 4 frames was randomly selected so that the final sequence length of the residual encoder was the same as that of the audio encoder. To prevent interference between adjacent frames, the negative samples were chosen to avoid overlapping with the positive ones. We used $\alpha = 0.1$, $\beta = 10$, and $\gamma = 1$ for loss weights. The output $\boldsymbol{r}'$ of the linear layer was used for prosodic representation, which had 64 channels.

PR_F0Energy: We used the energy and F0 spectrogram in FS2 as prosodic representation, which had 13 channels (1 for energy, 10 for CWT representation of F0, 2 for the mean and standard deviation of F0 before CWT). Compared with FS2, this model did not have the step of inverse continuous wavelet transform (iCWT), and the prosody prediction structure was changed from convolution to Transformer.

All the representation extraction models were trained on 4 $\times$ NVIDIA A100 GPUs with 5120000 audio samples or 320 sec, per GPU and the update frequency was set to 4, resulting

Table 1: *The objective evaluation results of natural speech and the speech generated by different models on the test set. Here, "Corr" means correlation.*

	F0 RMSE(Hz)	F0 CORR(%)	MSD(dB)	Duration Corr(%)
FS2	30.713	68.7	3.564	71.1
PR_F0Energy	30.373	69.0	3.548	70.5
S4LPR_V	28.797	70.7	3.379	73.7
S4LPR_LPC	28.172	72.2	3.391	73.3
S4LPR_MT	**27.618**	**73.0**	**3.378**	**73.9**

Table 2: *The naturalness mean opinion scores (MOS) of different models with 95% confidence intervals.*

Model	GT	FS2	S4LPR_MT
MOS	4.37 ± 0.09	4.03 ± 0.09	$\mathbf{4.14 \pm 0.09}$

in a total batch size of 5120 sec, or 1.4h. The model trained for about 5 days with 200k updates. The speech synthesis models were trained on 1 NVIDIA V100 GPU with a batch size of 20000 frames or 200 sec. All the models were trained with 200k updates for 2-3 days. We synthesized waveforms for evaluation with a pre-trained HiFi-GAN vocoder on the same dataset as the synthesis model.

3.2. Objective evaluation

For objective evaluation, the MFCC features extracted from the predicted audio are aligned with the MFCC features of the corresponding natural speech by the dynamic time warping (DTW) algorithm. Through the alignment path of DTW, F0 features were aligned as well, and then F0 correlation and F0 RMSE were calculated. We also aligned the predicted Mel-spectrogram with ground-truth directly using the DTW algorithm, and calculated the Mel-spectrogram distortion (MSD) as an objective evaluation metric. To calculate duration correlation, the forced alignment algorithm was first applied to align the synthesized speech with the phoneme sequences. Then we obtain the syllable duration from the correspondence between syllables and phonemes and calculate the correlation coefficient with recordings. The results are shown in Table 1. We can see that our three S4LPR models outperformed the other two models with human-designed intermediate prosodic features, and the S4LPR_MT model achieved the best objective performance.

3.3. Subjective evaluation

A mean opinion score (MOS) test was conducted to measure the naturalness of S4LPR_MT model and the FastSpeech2 baseline. 20 long sentences ($>=$ 30 characters) were randomly chosen from the test set and were synthesized by different models[1]. The MOS scale was from 1 to 5 (with intervals of 0.5), when 1 indicated completely unnatural and 5 indicated completely natural. 10 native Chinese listeners took part in the test. The speech recovered from the ground-truth Mel-spectrograms with the same HiFi-GAN vocoder was also evaluated for reference. The results are shown in Table 2. We can see that our proposed S4LPR_MT model achieved significantly higher naturalness than the FastSpeech2 baseline.

Three preference tests on naturalness were further conducted to compare the difference using different prosodic representations. At least 10 native Chinese listeners took part in each test and the results are shown in Table 3. We can see that S4LPR_MT achieved significantly better naturalness ($p < 0.05$) than the PR_F0Energy model using human-designed F0 and energy features. However, the naturalness differences among the three S4LPR models were insignificant.

[1] Audio samples can be found at `https://ttsbylzc.github.io/ttsdemo202303/`.

Table 3: *Average preference scores on naturalness among different models, where N/P means "no preference" and p denotes the p-value of a t-test between two models. System B is "S4LPR_MT".*

System A	Prefer A(%)	N/P(%)	Prefer B(%)	p
PR_F0Energy	30.4	28.3	**41.3**	0.047
S4LPR_V	33.9	33.5	32.6	0.813
S4LPR_LPC	34.6	25.8	39.6	0.370

Table 4: *The objective evaluation results of natural speech and the speech generated by different models on the test set. Here, "Corr" means correlation. The models labeled with * denote that the latent representation and duration were extracted from the recordings of test utterances instead of being predicted from text.*

	F0 RMSE(Hz)	F0 CORR(%)	MSD(dB)	Duration Corr(%)
FS2*	**6.511**	**98.7**	2.284	94.7
PR_F0Energy*	6.846	98.5	2.239	94.7
S4LPR_V*	9.259	97.3	2.224	**96.4**
S4LPR_LPC*	7.713	98.1	2.145	96.0
S4LPR_MT*	8.010	98.0	**2.144**	96.0
GT	6.516	98.7	0	97.8

3.4. Analysis

To analyze the description ability of different prosodic representations, we synthesized audio with the latent representations and the durations extracted from the recordings of test utterances instead of being predicted from the text. The results are shown in Table 4. When using features extracted from audio, the FS2 model and PR_F0Energy model had a slight advantage over the proposed method on F0 metrics. The duration correlations of the self-supervisedly learnt representations were all better than the human-designed features, despite of the lower frame rate. The representations given by the vanilla wav2vec 2.0 were the most accurate on duration, probably because these representations contain the most textual content. The proposed S4LPR_MT achieved the best MSD and the result was close to that of S4LPR_LPC, probably because the removal of prosody-irrelevant information benefits prediction of Mel-spectrograms.

4. Conclusions

In this study, we propose a speech synthesis method based on self-supervisedly learnt prosodic representations. The representation extraction models are pre-trained on a large-scale dataset in a self-supervised manner. A speech synthesis model based on FastSpeech2 and PnGBERT is designed to integrate the learnt prosodic representations. Experimental results show that our proposed method achieved better objective and subjective performance than the baseline FastSpeech2 model. To learn prosodic representation with other self-supervised schemes, eg. data2vec, will be the task of our future work.

5. References

[1] 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 *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.

[2] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T. Liu, "FastSpeech 2: Fast and high-quality end-to-end text to speech," in *9th International Conference on Learning Representations (ICLR)*, 2021.

[3] A. v. d. 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 preprint arXiv:1609.03499*, 2016.

[4] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis," *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[5] X. Tan, T. Qin, F. Soong, and T.-Y. Liu, "A survey on neural speech synthesis," *arXiv preprint arXiv:2106.15561*, 2021.

[6] A. Łańcucki, "FastPitch: Parallel text-to-speech with pitch prediction," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6588–6592.

[7] C. Torrence and G. P. Compo, "A practical guide to wavelet analysis," *Bulletin of the American Meteorological society*, vol. 79, no. 1, pp. 61–78, 1998.

[8] 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 *International Conference on Machine Learning*. PMLR, 2018, pp. 5180–5189.

[9] 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 *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6945–6949.

[10] Z. Hodari, A. Moinet, S. Karlapati, J. Lorenzo-Trueba, T. Merritt, A. Joly, A. Abbas, P. Karanasou, and T. Drugman, "CAMP: a two-stage approach to modelling prosody in context," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6578–6582.

[11] G. Zhang, Y. Qin, D. Tan, and T. Lee, "Applying the information bottleneck principle to prosodic representation learning," in *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA, 2021, pp. 3156–3160.

[12] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, vol. 33, pp. 12 449–12 460, 2020.

[13] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "data2vec: A general framework for self-supervised learning in speech, vision and language," in *International Conference on Machine Learning*. PMLR, 2022, pp. 1298–1312.

[14] S. Yang, P. Chi, Y. Chuang, C. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G. Lin, T. Huang, W. Tseng, K. Lee, D. Liu, Z. Huang, S. Dong, S. Li, S. Watanabe, A. Mohamed, and H. Lee, "SUPERB: speech processing universal performance benchmark," in *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA, 2021, pp. 1194–1198.

[15] Y. Lim, N. Kim, S. Yun, S. Kim, and S.-I. Lee, "A preliminary study on wav2vec 2.0 embeddings for text-to-speech," in *International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2021, pp. 343–347.

[16] H. Siuzdak, P. Dura, P. van Rijn, and N. Jacoby, "WavThruVec: Latent speech representation as intermediate features for neural speech synthesis," in *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA, 2022, pp. 833–837.

[17] A. Polyak, Y. Adi, J. Copet, E. Kharitonov, K. Lakhotia, W.-N. Hsu, A. Mohamed, and E. Dupoux, "Speech resynthesis from discrete disentangled self-supervised representations," in *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2021, pp. 3615–3619.

[18] L. Chen, Y. Deng, X. Wang, F. K. Soong, and L. He, "Speech BERT embedding for improving prosody in neural TTS," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6563–6567.

[19] Y. Hu, C. Zhang, J. Shi, J. Lian, M. Ostendorf, and D. Yu, "ProsodyBERT: Self-supervised prosody representation for style-controllable TTS," 2023. [Online]. Available: https://openreview.net/forum?id=7wk9PqiiW2D

[20] Y. Jia, H. Zen, J. Shen, Y. Zhang, and Y. Wu, "PnG BERT: augmented BERT on phonemes and graphemes for neural TTS," in *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA, 2021, pp. 151–155.

[21] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *5th International Conference on Learning Representations (ICLR) - Conference Track Proceedings*, 2017, pp. 1–13.

[22] L. Rabiner and R. Schafer, *Theory and applications of digital speech processing*. Prentice Hall Press, 2010.

[23] 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.

[24] Z. Liu, N. Wu, Y. Zhang, and Z. Ling, "Integrating discrete word-level style variations into non-autoregressive acoustic models for speech synthesis," *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 5508–5512, 2022.

[25] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019, pp. 4171–4186.

[26] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, "fairseq: A fast, extensible toolkit for sequence modeling," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019, pp. 48–53.