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

Rich Prosody Diversity Modelling with Phone-level Mixture Density Network

*Chenpeng Du, Kai Yu**

MoE Key Lab of Artificial Intelligence, AI Institute
X-LANCE Lab, Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai, China
{duchenpeng, kai.yu}@sjtu.edu.cn

Abstract

Generating natural speech with a diverse and smooth prosody pattern is a challenging task. Although random sampling with phone-level prosody distribution has been investigated to generate different prosody patterns, the diversity of the generated speech is still very limited and far from what can be achieved by humans. This is largely due to the use of uni-modal distribution, such as single Gaussian, in the prior works of phone-level prosody modelling. In this work, we propose a novel approach that models phone-level prosodies with GMM based mixture density network (GMM-MDN). Experiments on the LJSpeech dataset demonstrate that phone-level prosodies can precisely control the synthetic speech and GMM-MDN can generate a more natural and smooth prosody pattern than a single Gaussian. Subjective evaluations further show that the proposed approach not only achieves better naturalness, but also significantly improves the prosody diversity in synthetic speech without the need of manual control.

Index Terms: mixture density network, Gaussian mixture model, prosody modelling, speech synthesis

1. Introduction

Neural text-to-speech(TTS) synthesis models with sequence-to-sequence architecture [1, 2, 3] can be applied to generate naturally sounding speech. Recently, non-autoregressive TTS models such as FastSpeech [4] and FastSpeech2 [5] are proposed for fast generation speed without frame-by-frame generation.

Besides the progress of acoustic modelling, prosody modelling is also widely investigated. Utterance level prosody modelling in TTS is proposed in [6], in which an utterance-level prosody embedding is extracted from a reference speech for controlling the prosody of TTS output. [7] factorizes the prosody embedding with several global style tokens(GST). Variational auto-encoder(VAE) is used for prosody modelling in [8], which enables us to sample various prosody embeddings from the standard Gaussian prior in inference stage.

Utterance-level prosody is hard to precisely control the synthetic speech due to its coarse granularity. Therefore, phone-level prosody modelling is also analyzed in recent works. [9] extracts prosody information from acoustic features and uses an attention module to align them with each phoneme encoding. [10] directly models phone-level prosody with a VAE, thus improving the stability compared with [9]. Hierarchical and quantized versions of VAE for phone-level prosody modelling is also investigated in [11, 12, 13, 14], which improves the interpretability and naturalness in synthetic speech.

However, most of the prior works for phone-level prosody modelling assumes that the distribution of phone-level prosodies is uni-modal distribution, such as a single Gaussian, which is not reasonable enough and hence can only provide limited diversity. Actually, phone-level prosodies are highly diverse even for the same context, hence it is natural to apply multi-modal distribution. In traditional ASR systems, one of the most dominant techniques is HMM-GMM [15, 16, 17], in which the distribution of acoustic features for each HMM state is modeled with a GMM. Similarly, GMM is also used to model acoustic features in traditional statistical parametric speech synthesis(SPSS) [18, 19], thus improving the voice quality.

Inspired by these prior works, we propose a novel approach that models phone-level prosodies with GMM based mixture density network (GMM-MDN) [20]. We use a prosody extractor to extract phone-level prosody embeddings from ground-truth mel-spectrograms and use a prosody predictor as the MDN to predict the GMM distribution of the embeddings. In inference stage, the prosody of each phoneme is randomly sampled from the predicted GMM distribution for generating speech with diverse prosodies.

Our experiments on the LJSpeech [21] dataset demonstrate that phone-level prosodies can precisely control the synthetic speech and GMM-MDN can generate more natural and smooth prosody pattern than a single Gaussian. The subjective evaluations suggest that our method not only achieves a better naturalness, but also significantly improves the prosody diversity in synthetic speech without the need of manual control.

In the rest of this paper, we first review the MDN in Section 2 and introduce the proposed model in Section 3. Section 4 gives experiments comparison and results analysis, and Section 5 concludes the paper.

2. Mixture Density Network

Mixture density network [20] is defined as the combined structure of a neural network and a mixture model. We focus on GMM based MDN in this work to predict the parameters of the GMM distribution, including the means μ_i, variances σ_i^2, and mixture weights w_i. It should be noted that the sum of the mixture weights is constrained to 1, which can be achieved by applying a Softmax function, formalized as

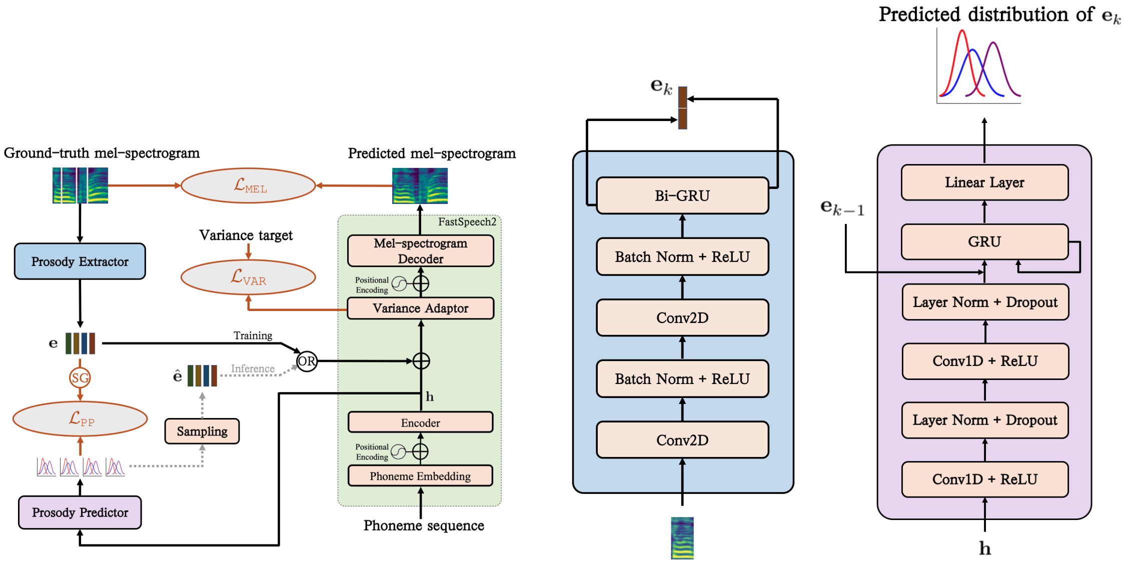
$$w_i = \frac{\exp(\alpha_i)}{\sum_{j=1}^{M} \exp(\alpha_j)} \tag{1}$$

where M is the number of Gaussian components and α_i is the corresponding neural network output. The mean and variance of Gaussian components are presented as

$$\begin{aligned} \mu_i &= m_i & (2) \\ \sigma_i^2 &= \exp(v_i) & (3) \end{aligned}$$

* Corresponding author.

http://dx.doi.org/10.21437/Interspeech.2021-802



(a) Overal architecture based on FastSpeech2

(b) Prosody extractor

(c) Prosody predictor

Figure 1: *GMM-based phone-level prosody modelling. "SG" represents the stop gradient operation. "OR" selects the extracted "ground-truth"* $\mathbf{e}$ *in training and the sampled* $\hat{\mathbf{e}}$ *in inference. We use red lines for loss calculation and dash lines for sampling.*

where m_i and v_i are the neural network outputs corresponding to the mean and variance of the i-th Gaussian component. Equation (3) constrains the σ_i^2 to be positive.

The criterion for training the MDN is the negative log-likelihood of the observation $\mathbf{y}$ given its input $\mathbf{x}$. Here we can formulate the loss function as

$$\begin{aligned}\mathcal{L}_{\text{MDN}} &= -\log p\left(\mathbf{y};\mathbf{x}\right)\\ &= -\log\left(\sum_{i=1}^{M} w_i \cdot \mathcal{N}\left(\mathbf{y};\mu_i,\sigma_i^2\right)\right)\end{aligned} \tag{4}$$

Therefore, given the input $\mathbf{x}$, the mixture density network is optimized to predict GMM parameters w_i, μ_i and σ_i that maximize the likelihood of $\mathbf{y}$.

3. GMM-MDN for Phone-Level Prosody Modelling

The TTS model in this paper is based on the recent proposed FastSpeech2[5], where the input phoneme sequence is first converted into a hidden state sequence $\mathbf{h}$ by the encoder and then passed through a variance adaptor and a decoder for predicting the output mel-spectrogram. Compared with the original FastSpeech[4], FastSpeech2 is optimized to minimize the mean square error(MSE) $\mathcal{L}_{\text{MEL}}$ between the predicted and the ground-truth mel-spectrograms, instead of applying a teacher-student training. Moreover, the duration target is not extracted from the attention map of an autoregressive teacher model, but from the forced alignment of speech and text. Additionally, [5] condition the prediction of mel-spectrogram on the variance information such as pitch and energy with a variance adaptor. The adaptor is trained to predict the variance information with an MSE loss $\mathcal{L}_{\text{VAR}}$.

In this work, we propose a novel approach that models phone-level prosodies with a GMM-MDN in the FastSpeech2-based TTS system. Accordingly, we introduce a prosody extractor and a prosody predictor, whose architecture and training strategy are described below.

3.1. Overall architecture

The overall architecure of the proposed system is shown in Figure 1(a). In the training stage, the prosody embeddings

$$\mathbf{e} = [\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_K] \tag{5}$$

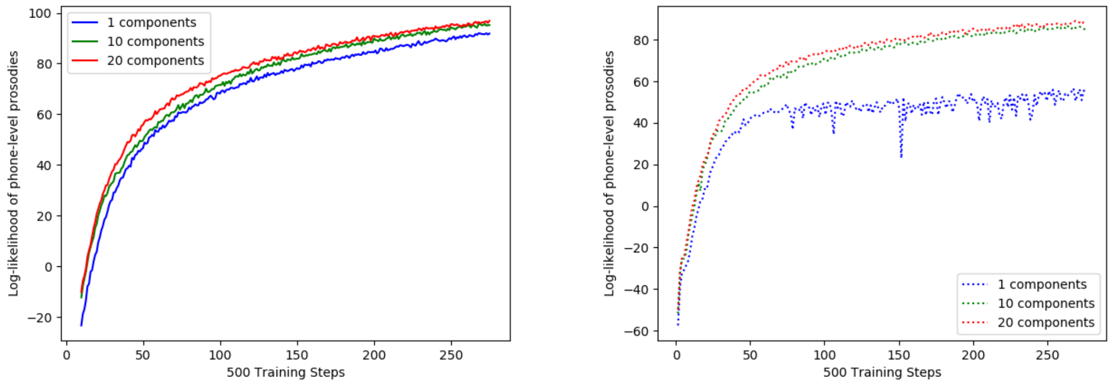
are extracted for all the K phonemes by the prosody extractor from the corresponding mel-spectrogram segment. It is then projected and added to the corresponding hidden state sequence $\mathbf{h}$ in order to better reconstruct the mel-spectrogram. We use $\mathbf{e}_k$ to represent the prosody embedding for the k-th phoneme. The distribution of $\mathbf{e}_k$ is assumed to be a GMM whose parameters are predicted by an MDN. Here, the GMM-MDN is the prosody predictor. The prosody predictor autoregressively predicts the GMM distributions of the prosody embeddings. In inference stage, we sample the $\hat{\mathbf{e}}_k$ from the predicted distribution for each phoneme, so that we can generate speech with diverse prosodies.

3.2. Prosody extractor

The detailed architecture of the prosody extractor is shown in Figure 1(b). It contains 2 layers of 2D convolution, each followed by a batch normalization layer and a ReLU activation function. A bidirectional GRU is designed after the above modules. The concatenated forward and backward states from the GRU layer is the output of the prosody extractor, which is referred to as the prosody embedding of the phoneme.

3.3. Prosody predictor

The GMM-MDN in this work is the prosody predictor whose detailed architecture is demonstrated in Figure 1(c). The hidden state $\mathbf{h}$ of the input phoneme sequence is passed through 2



(a) LJSpeech training set

(b) LJSpeech test set

Figure 2: *Log-likelihood curves of the extracted phone-level prosodies with different numbers of Gaussian components on LJSpeech and LibriTTS.*

layers of 1D convolution, each followed by a ReLU, layer normalization and dropout layer. The output of the above modules is then concatenated with the previous prosody embedding and sent to a GRU. The GRU is designed to condition the prediction of the current prosody distribution on the previous prosodies. Then we project the GRU output to obtain $w_{k,i}$, $m_{k,i}$ and $v_{k,i}$, which is then transformed to the GMM parameters according to Equation (1) - (3).

Equation (4) formulates the training criterion for an MDN, which is the negative log-likelihood of the observations. Here, the observations are the prosody embeddings $\mathbf{e}$, so we obtain the loss function for training the prosody predictor

$$\begin{aligned}\mathcal{L}_{\texttt{PP}} &= \sum_{k=1}^{K} -\log p\left(\mathbf{e}_k; \mathbf{e}_{<k}, \mathbf{h}\right) \\ &= \sum_{k=1}^{K} -\log\left(\sum_{i=1}^{M} w_{k,i} \cdot \mathcal{N}\left(\mathbf{e}_k; \mu_{k,i}, \sigma_{k,i}^2\right)\right)\end{aligned} \tag{6}$$

where $w_{k,i}$, $\mu_{k,i}$ and $\sigma_{k,i}$ are the Gaussian parameters of the i-th component. The parameters are predicted given $\mathbf{h}$ and $\mathbf{e}_{<k}$.

3.4. Training criterion

The prosody extractor and the prosody predictor are both jointly trained with the FastSpeech2 architecture. The overall architecture is optimized with the loss function

$$\begin{aligned}\mathcal{L} &= \beta \cdot \mathcal{L}_{\texttt{PP}} + \mathcal{L}_{\texttt{FastSpeech2}} \\ &= \beta \cdot \mathcal{L}_{\texttt{PP}} + \left(\mathcal{L}_{\texttt{MEL}} + \mathcal{L}_{\texttt{VAR}}\right)\end{aligned} \tag{7}$$

where $\mathcal{L}_{\texttt{PP}}$ is defined in Equation (6), $\mathcal{L}_{\texttt{FastSpeech2}}$ is the loss function of FastSpeech2 which is the sum of variance prediction loss $\mathcal{L}_{\texttt{VAR}}$ and mel-spectrogram reconstruction loss $\mathcal{L}_{\texttt{MEL}}$ as described in [5], and β is the relative weight between the two terms. It should be noted that we use a stop gradient operation on $\mathbf{e}$ in calculating the $\mathcal{L}_{\texttt{PP}}$, so the prosody extractor is not optimized with $\mathcal{L}_{\texttt{PP}}$ directly.

4. Experiments and Results

4.1. Experimental setup

LJSpeech [21] is an English dataset, containing about 24 hours speech recorded by a female speaker. We randomly leave out 250 utterances for testing.

All the speech data in this work is resampled to 16kHz for simplicity. The mel-spectrograms are extracted with 50ms window, 12.5ms frame shift, 1024 FFT points and 320 mel-bins. Before training TTS, we compute the phoneme alignment of the training data with an HMM-GMM ASR model trained on Librispeech [22], and then extract the duration of each phoneme from the alignment for TTS training.

The β in Equation (7) is set to 0.02. An Adam optimizer [23] is used for TTS training in conjunction with a noam learning rate scheduler [24]. The output mel-spectrogram is converted to the waveform with a MelGAN [25] vocoder, which is trained on the same training set.

In addition to the proposed system PLP-GMM, we also build two other systems ULP and PLP-SG with different prosody modelling methods. 1) ULP: A recent popular approach for utterance-level prosody (ULP) modelling is conditioning the synthesis on a latent variable from VAE [8]. The dimensionality of the latent variable is set to 128. In the training stage, the latent variable is sampled from the posterior for each utterance. In the inference stage, the latent variable is sampled from the prior distribution, which is a standard Gaussian $\mathcal{N}(0, I)$. 2) PLP-SG: For precise prosody modelling, we apply phone-level prosodies (PLP) in this work. One basic method of PLP modelling is using a single Gaussian [10, 12]. 3) PLP-GMM: As is described in Section 3, the proposed system models the phone-level prosodies with a GMM based MDN, which means the output of the prosody predictor is the parameters of GMMs. The other configurations of PLP-GMM are the same as PLP-SG.

4.2. The necessity of using phone-level prosodies

Firstly, we verify whether using the extracted ground-truth phone-level prosodies $\mathbf{e}$ is better than using utterance-level prosodies in reconstruction. We reconstruct the test set with PLP-GMM and ULP, which is guided by the extracted phone-level prosody $\mathbf{e}$ and by the utterance-level prosody sampled from the latent posterior respectively.

Mel-cepstral distortion (MCD) [26] is an objective measure of the distance between two sequences of mel-cepstral coefficients. We extract 25 dimensional mel-cepstral coefficients with 5ms frame shift [27] from both the synthetic speech and the corresponding ground-truth speech on the test set for computing the MCD. The results are demonstrated in Table 1, where a lower MCD represents a better reconstruction performance. It can be observed that using the extracted phone-level prosodies achieves lower MCD than the utterance-level baselines. This is natural because phone-level prosody representations contain much more information about how each phoneme is pronounced than an utterance-level representation. Therefore, it is necessary to use phone-level prosodies in TTS systems for precise prosody controlling.

Table 1: *Reconstruction performance (MCD) on the test set.*

Prosody Condition	MCD
Utterance-level	5.22
Phone-level	3.38

4.3. The number of Gaussian components for phone-level prosody modelling

In this section, we try to figure out how many Gaussian components are needed to model the distribution of the extracted phone-level prosodies $\mathbf{e}$. We plot the log-likelihood curves on both the training set and the test set with several different numbers of Gaussian components in Figure 2.

We can find that the log-likelihood gap between the training and test set in the single Gaussian model is larger than that in the Gaussian mixture models. Moreover, the log-likelihood curves of Gaussian mixture models are much higher than the single Gaussian model in both the training and test set. Therefore, we can conclude that the distribution of phone-level prosodies is multi-modal and it should be modeled with Gaussian mixtures.

Additionally, increasing the number of components also provides higher log-likelihood, because the more components enable the model to simulate more complicated distribution. However, when we double the number of the components from 10 to 20, the improvement is very limited. Therefore, we do not further increase the number of components, and use 20 components in the following GMM-based systems.

4.4. Prosody diversity

According to the investigation above, we train the proposed system PLP-GMM with 20 Gaussian components. In the inference stage, the Gaussian mixture distributions of phone-level prosodies are predicted, from which the phone-level prosodies $\hat{\mathbf{e}}$ are sampled. The synthesis is then guided by the sampled prosodies $\hat{\mathbf{e}}$. [1]

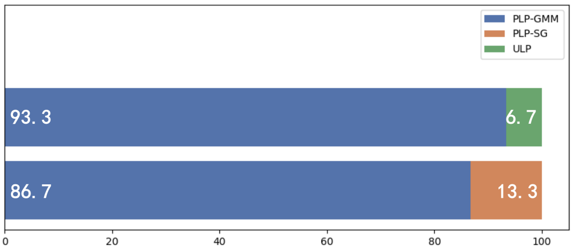


Figure 3: *AB preference test in terms of prosody diversity.*

We compare PLP-GMM with two baseline systems ULP and PLP-SG in terms of prosody diversity. We synthesize the utterances in the test set 3 times with various sampled prosodies. We perform AB preference tests where two groups of synthetic speech from two different TTS systems are presented and the listeners need to select the better one in terms of prosody diversity. The results are shown in Figure 3. We can find that PLP-GMM provides better prosody diversity in the synthetic speech than both ULP and PLP-SG. This can be easily explained by the fact that a sequence of phone-level embeddings depicts the prosody more precisely than an utterance-level embedding and the fact that Gaussian mixtures can better model the phone-level prosody embeddings than a single Gaussian.

4.5. Naturalness

We also evaluate the naturalness of the above systems with a MUSHRA test, in which the listeners are asked to rate each utterance on a scale of 0 to 100. The speech converted back from the ground-truth mel-spectrogram with the vocoder is also rated in the test, which is denoted as GT. The results are reported in Figure 4. It can be observed that PLP-GMM synthesizes speech with higher naturalness score compared with PLP-SG because of the better phone-level prosody modelling. We can also find that ULP generates speech with similar naturalness as PLP-GMM, which can also be easily explained. In ULP, no phone-level prosody is explicitly considered. Hence, the synthetic speech contains an averaged phone-level prosodies, whose naturalness is comparable to the sampled phone-level prosodies in PLP-GMM. Finally, we notice that the median of PLP-GMM is also slightly higher than ULP, but both the medians of PLP-GMM and ULP are lower than GT.

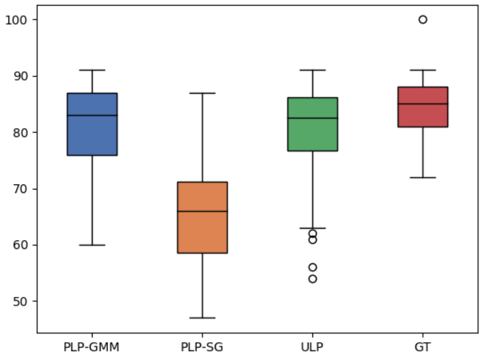


Figure 4: *MUSHRA test in terms of naturalness.*

5. Conclusions

In this work, we propose a novel approach that uses a GMM-MDN to model the phone-level prosodies. Our experiments first prove that using the extracted phone-level prosodies can better reconstruct the speech than using utterance-level prosodies. Then we find that the log-likelihood of phone-level prosodies increases when more Gaussian components are used, indicating that GMM-MDN can generate more natural and smooth prosody pattern than a single Gaussian. The subjective evaluations suggest that our method not only achieves a better naturalness, but also significantly improves the prosody diversity in synthetic speech without the need of manual control.

6. Acknowledgements

This study was supported by Shanghai Municipal Science and Technology Major Project No.2021SHZDZX0102 and Xinhua News Agency No.SKLMCPTS2020003.

[1] Audio examples are available here `https://cpdu.github.io/gmm_prosody_examples`.

7. References

[1] Y. Wang, R. J. Skerry-Ryan, D. Stanton *et al.*, "Tacotron: Towards end-to-end speech synthesis," in *Proc. ISCA 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. Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning wavenet on MEL spectrogram predictions," in *Proc. IEEE ICASSP*, 2018, pp. 4779–4783.

[3] N. Li, S. Liu, Y. Liu, S. Zhao, M. Liu, and M. Zhou, "Close to human quality TTS with transformer," *arXiv preprint arXiv:1809.08895*, 2018.

[4] 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. NIPS*, 2019, pp. 3165–3174.

[5] Y. Ren, C. Hu, T. Qin, S. Zhao, Z. Zhao, and T. Liu, "Fastspeech 2: Fast and high-quality end-to-end text-to-speech," *arXiv preprint arXiv:2006.04558*, 2020.

[6] R. J. Skerry-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor, R. J. Weiss, R. Clark, and R. A. Saurous, "Towards end-to-end prosody transfer for expressive speech synthesis with tacotron," in *Proc. ICML*, 2018, pp. 4700–4709.

[7] Y. Wang, D. Stanton, Y. Zhang, R. J. Skerry-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. 5167–5176.

[8] K. Akuzawa, Y. Iwasawa, and Y. Matsuo, "Expressive speech synthesis via modeling expressions with variational autoencoder," in *Proc. ISCA Interspeech*, 2018, pp. 3067–3071.

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

[10] V. Klimkov, S. Ronanki, J. Rohnke, and T. Drugman, "Fine-grained robust prosody transfer for single-speaker neural text-to-speech," in *Proc. ISCA Interspeech*, 2019, pp. 4440–4444.

[11] 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. IEEE ICASSP*, 2020, pp. 6264–6268.

[12] G. Sun, Y. Zhang, R. J. Weiss, Y. Cao, H. Zen, A. Rosenberg, B. Ramabhadran, and Y. Wu, "Generating diverse and natural text-to-speech samples using a quantized fine-grained VAE and autoregressive prosody prior," in *Proc. IEEE ICASSP*, 2020, pp. 6699–6703.

[13] Y. Hono, K. Tsuboi, K. Sawada, K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "Hierarchical multi-grained generative model for expressive speech synthesis," in *Proc. ISCA Interspeech*, 2020, pp. 3441–3445.

[14] C. M. Chien and H. Y. Lee, "Hierarchical prosody modeling for non-autoregressive speech synthesis," in *Proc. IEEE SLT*, 2021, pp. 446–453.

[15] P. C. Woodland, C. Leggetter, J. Odell, V. Valtchev, and S. Young, "The development of the 1994 htk large vocabulary speech recognition system," in *Proceedings ARPA workshop on spoken language systems technology*, 1995, pp. 104–109.

[16] J. Gauvain and C. Lee, "Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains," *IEEE Trans. Speech Audio Process.*, vol. 2, no. 2, pp. 291–298, 1994.

[17] M. J. F. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition," *Computer speech & language*, vol. 12, no. 2, pp. 75–98, 1998.

[18] H. Zen and A. W. Senior, "Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis," in *Proc. IEEE ICASSP*, 2014, pp. 3844–3848.

[19] X. Wang, S. Takaki, and J. Yamagishi, "An autoregressive recurrent mixture density network for parametric speech synthesis," in *Proc. IEEE ICASSP*, 2017, pp. 4895–4899.

[20] C. M. Bishop, "Mixture density networks," *Aston University*, 1994.

[21] K. Ito, "The lj speech dataset," https://keithito.com/LJ-Speech-Dataset/, 2017.

[22] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *Proc. IEEE ICASSP*, 2015, pp. 5206–5210.

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.

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

[25] K. Kumar, R. Kumar, T. de Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. de Brébisson, Y. Bengio, and A. C. Courville, "Melgan: Generative adversarial networks for conditional waveform synthesis," in *Proc. NIPS*, 2019, pp. 14 881–14 892.

[26] R. Kubichek, "Mel-cepstral distance measure for objective speech quality assessment," in *Proceedings of IEEE Pacific Rim Conference on Communications Computers and Signal Processing*, vol. 1, 1993, pp. 125–128.

[27] J. Kominek, T. Schultz, and A. W. Black, "Synthesizer voice quality of new languages calibrated with mean mel cepstral distortion," in *Proc. ISCA SLTU*, 2008, pp. 63–68.