



VoiceTailor: Lightweight Plug-In Adapter for Diffusion-Based Personalized Text-to-Speech

Heeseung Kim¹, Sang-gil Lee², Jiheum Yeom¹, Che Hyun Lee¹, Sungwon Kim², Sungroh Yoon^{*1,3}

¹Data Science and AI Lab, ECE, Seoul National University (SNU), Korea

²NVIDIA

³AIIS, ASRI, INMC, ISRC, and Interdisciplinary Program in AI, SNU, Korea

{gmltmd789, quilava1234, saga1214, sryoon}@snu.ac.kr, {sanggill, sungwon}@nvidia.com

Abstract

We propose VoiceTailor, a parameter-efficient speaker-adaptive text-to-speech (TTS) system, by equipping a pre-trained diffusion-based TTS model with a personalized adapter. VoiceTailor identifies pivotal modules that benefit from the adapter based on a weight change ratio analysis. We utilize Low-Rank Adaptation (LoRA) as a parameter-efficient adaptation method and incorporate the adapter into pivotal modules of the pre-trained diffusion decoder. To achieve powerful adaptation performance with few parameters, we explore various guidance techniques for speaker adaptation and investigate the best strategies to strengthen speaker information. VoiceTailor demonstrates comparable speaker adaptation performance to existing adaptive TTS models by fine-tuning only 0.25% of the total parameters. VoiceTailor shows strong robustness when adapting to a wide range of real-world speakers, as shown in the demo¹.

Index Terms: text-to-speech (TTS), adaptive TTS, parameter-efficient TTS, diffusion, Low-Rank Adaptation (LoRA)

1. Introduction

Recent advancements in deep generative models have led to improvements in adaptive text-to-speech (TTS), enabling models to generate a target speaker's voice from a given transcript and reference speech [1, 2, 3]. Zero-shot approach [1, 3, 4, 5, 6] for adaptive TTS eliminates the need for extra fine-tuning on reference audio for speaker adaptation. Despite its advantage of no further training, this approach generally requires large speech corpus during training to achieve high speaker similarity, and is comparatively less robust against unique out-of-distribution voices commonly encountered in real-world scenarios.

One-shot approach, an alternative type of adaptive TTS, constructs personalized TTS by fine-tuning pre-trained multi-speaker TTS models with few reference speeches of target speaker [1, 7, 8, 9, 10, 11, 12]. To efficiently adapt to the target speaker, several studies fine-tuned a subset of the model's parameters [7, 8, 10, 11, 12], or leveraged adapter-based fine-tuning techniques [9] such as Low-Rank Adaptation (LoRA) [13] or prefix-tuning [14], which only fine-tune the parameters of newly integrated adapters. However, these works often fail to generate speech with high speaker similarity due to the limitations of the generative models used as decoder and typically require more than a minute of speech data for fine-tuning.

Recently, inspired by successes of diffusion-based generative model [15] on fine-tuning-based personalized generation tasks [16], diffusion-based one-shot TTS models have been proposed [2, 17]. They leverage the diffusion model's adaptation

performance to achieve high speaker similarity in personalized TTS task with as short as 5 to 10 seconds of reference speech. However, in contrast to other one-shot approaches, these works fine-tune all model parameters, resulting in parameter inefficiency.

In this work, we introduce VoiceTailor, a parameter-efficient adaptive TTS model that requires fine-tuning only a subset of parameters from a diffusion-based pre-trained TTS model. We utilize a diffusion-based pre-trained TTS model and adopt a fine-tuning methodology following UnitSpeech [17]. Inspired by the approaches in [18, 19], we analyze the change ratio in the weights of each module in the model before and after fine-tuning and identify that attention modules play a crucial role in speaker adaptation. Based on this observation, VoiceTailor carefully integrates LoRA into the effective attention modules in the model and fine-tunes only the injected low-rank matrices for adaptation.

We demonstrate that VoiceTailor achieves speaker adaptation performance comparable to the fully fine-tuned one-shot baseline by plugging in the small adapter with 0.25% of the total parameters of the pre-trained model, which occupies approximately 1.3 MB of storage space. In addition, we systematically analyze the impact of various design choices and hyperparameters during the parameter-efficient adaptation stage. Furthermore, we investigate the best strategy from various guidance techniques in the inference stage. We illustrate VoiceTailor's robust performance in real-world scenarios by presenting a variety of samples, including those adapted for real-world speakers, on our demo page. Our contributions are as follows:

- To the best of our knowledge, this is the first work that systematically incorporates LoRA for diffusion-based speaker adaptive TTS that achieves high speaker similarity.
- VoiceTailor significantly reduces cost of adapting TTS to new speaker using 10 seconds of untranscribed speech with approximately 15 seconds of training time on a single GPU by utilizing 0.25% of the model parameters.
- We compare and analyze various methods to enhance speaker information using LoRA modules and speaker classifier-free guidance and investigate the optimal strategy.

2. Method

We introduce VoiceTailor, a personalized TTS model utilizing LoRA to address the parameter inefficiency prevalent in existing diffusion-based one-shot TTS approaches. VoiceTailor captures the target speaker's characteristics through LoRA fine-tuning and a speaker embedding extracted from a reference audio. We conduct a weight change ratio analysis of an existing model, UnitSpeech [17], and explore various methodologies to enhance the speaker information. Through careful injection of

* Corresponding Author

¹Demo: <https://voicetailor.github.io/>

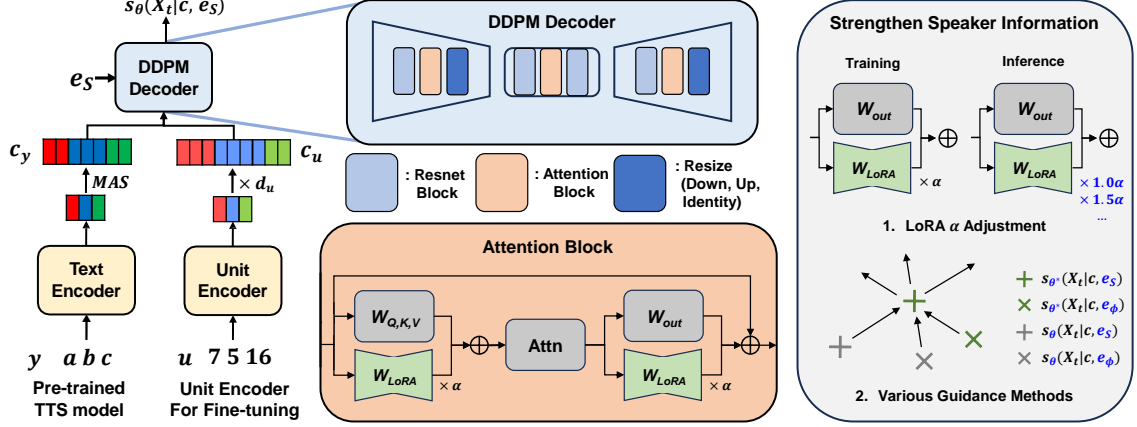


Figure 1: An overview of VoiceTailor depicting the LoRA adapters and techniques for strengthening speaker information.

LoRA weights from our analysis and selecting the optimal strategy for guidance technique, VoiceTailor achieves personalized TTS by fine-tuning as few as 0.25% of the model’s total parameters. A detailed overview of VoiceTailor is illustrated in Figure 1. UnitSpeech, our baseline model for the one-shot approach, is specified in Section 2.1. We describe details of the fine-tuning process using LoRA in Section 2.2. We introduce several strategies to strengthen the target speaker information when synthesizing personalized speech in Section 2.3.

2.1. UnitSpeech

In this work, we employ UnitSpeech [17], an adaptive speech synthesis model with powerful personalization capabilities, serving as the foundation for our one-shot TTS approach. UnitSpeech introduces a method to construct a personalized TTS model by fine-tuning a pre-trained, multi-speaker, diffusion-based TTS model with a short untranscribed speech sample.

The multi-speaker diffusion-based TTS model in UnitSpeech is based on Grad-TTS [20], which first defines a forward process that converts a mel-spectrogram X_0 to Gaussian noise $X_T \sim N(0, I)$. The forward process is defined using the pre-defined noise schedule β_t and the Wiener process W_t . The noisy mel-spectrogram X_t at timestep $t \in [0, T]$ in the forward process is computed as follows:

$$dX_t = -\frac{1}{2}X_t\beta_t dt + \sqrt{\beta_t}dW_t, \quad t \in [0, 1], \quad (1)$$

$$X_t = \sqrt{e^{-\int_0^t \beta_s ds}} X_0 + \sqrt{1 - e^{-\int_0^t \beta_s ds}} \epsilon_t. \quad (2)$$

Here, ϵ_t is the noise sampled from the standard normal distribution.

To sample the mel-spectrogram along the reverse trajectory of the previously defined process, it is necessary to utilize a score $s(X_t|c_y, e_S)$ that is conditioned on the text encoder output c_y and the speaker embedding e_S extracted from the pre-trained speaker encoder. UnitSpeech’s diffusion-based decoder θ is trained to predict the conditional score $s_\theta(X_t|c_y, e_S)$. The loss function for decoder pre-training and the formula of using the predicted score for sampling are as follows:

$$L = \mathbb{E}_{t, X_0, \epsilon_t} [\|(\sqrt{1 - e^{-\int_0^t \beta_s ds}} s_\theta(X_t|c_y, e_S) + \epsilon_t)\|_2^2], \quad (3)$$

$$X_{t-\Delta t} = X_t + \beta_t \left(\frac{1}{2} X_t + s_\theta(X_t|c_y, e_S) \right) \Delta t + \sqrt{\beta_t \Delta t} z_t, \quad (4)$$

where $z_t \sim N(0, I)$ is Gaussian noise.

UnitSpeech introduces a unit encoder to fine-tune the pre-trained diffusion decoder with untranscribed speech, eliminating the need for text input during the speaker adaptation process. The unit encoder is designed to replace the text encoder by receiving acoustic units (*i.e.*, self-supervised speech representations containing phonetic information [21]) as input. By substituting the text encoder with this pluggable unit encoder and training it with the same objective as the pre-trained decoder, UnitSpeech can receive unit inputs in addition to text inputs. This approach enables speaker adaptation by fine-tuning the decoder with the reference audio and its corresponding unit.

UnitSpeech integrates classifier-free guidance [22], a method for enhancing conditioning information in diffusion models, into the text encoder output c_y for accurate pronunciation. Unlike UnitSpeech, which solely applies classifier-free guidance to text conditions, we extend this approach to speaker embeddings e_S as well. While pre-training the multi-speaker TTS model, we introduce a learnable unconditional embedding e_ϕ and substitute e_S with e_ϕ with a probability of 25%. The resulting unconditional score obtained with e_ϕ is then utilized for speaker classifier-free guidance, as detailed in Section 2.3.

2.2. Parameter-Efficient Speaker Adaptation

To address the inefficiency of fine-tuning all parameters during speaker adaptation, we incorporate LoRA [13], a parameter-efficient adaptation technique. LoRA is a method that allows fine-tuning of the linear layer’s weight matrix by combining trainable low-rank decomposed matrices. Given a pre-trained weight $W \in \mathbb{R}^{d \times k}$ of the linear layer, LoRA augments it with $W + \alpha \cdot \Delta W = W + \alpha \cdot BA$, where the parameters $\Delta W := W_{LoRA}$ are fine-tuned with W being frozen. Here, $B \in \mathbb{R}^{d \times r}$, $A \in \mathbb{R}^{r \times k}$, α is the scaling factor of the adapter matrices, and r represents the rank. By using a significantly smaller value for the rank r compared to the dimensions d, k of the original matrix ($r \ll d, k$), LoRA enables adaptation with orders of magnitude fewer parameters. We denote the pre-trained model’s parameters as θ , and the parameters of the model with the fine-tuned adapter (W_{LoRA}) as θ^* .

Inspired by [18, 19], we first conduct speaker adaptation by fine-tuning all decoder parameters using UnitSpeech to explore which modules play a pivotal role in speaker adaptation. We measure the weight change ratio $\|\theta_i^* - \theta_i\| / \|\theta_i\|$ for each module θ_i before and after fine-tuning. Considering the prevalent application of LoRA to attention modules [13, 18], we mea-

sure the average change ratios of weight in the attention module and other modules within UnitSpeech’s diffusion decoder, obtaining values of 0.0282 and 0.0050, respectively. These results confirm that, similar to [18], the attention module is crucial in adaptation for one-shot diffusion-based TTS models. Consequently, we inject LoRA into the attention module and optimize only these parameters for speaker adaptation. During this fine-tuning process, we use the same objective used for DDPM decoder pre-training in UnitSpeech, as specified in Eq. 3.

2.3. Speaker Information Strengthening Strategies

The fine-tuned adapter, combined with the pre-trained multi-speaker TTS model, enables us to construct personalized TTS for the target speaker. In VoiceTailor, the speaker information is provided in two forms: the speaker embedding (e_S) and the pluggable LoRA weights (W_{LoRA}). To mitigate degradation in speaker adaptation performance due to decreased parameters, we explore various approaches for sampling to strengthen the target speaker’s information. We consider adjusting the scaling factor α of LoRA to a value greater than what is used during fine-tuning, and applying classifier-free guidance to both forms of information.

Adjustment of LoRA scaling factor α controls the intensity with which the adapter is added to the pre-trained model for speaker adaptation. By using a larger α during generation than the one used during training, we aim to provide stronger speaker information contained within the low-rank adapter.

Classifier-free guidance As there are two sources of speaker information (e_S and W_{LoRA}), we consider classifier-free guidance for each source. Given the score of fine-tuned model $s_{\theta^*}(X_t|c, e_S)$, we consider 3 candidates for the unconditional score s_{uncon} :

1. $s_{\theta^*}(X_t|c, e_\phi)$ can be obtained by replacing e_S with the unconditional embedding e_ϕ while maintaining the speaker information provided by W_{LoRA} .
2. $s_\theta(X_t|c, e_S)$ can be obtained from the pre-trained model θ by removing W_{LoRA} and keeping e_S as input.
3. $s_\theta(X_t|c, e_\phi)$ can also be used as s_{uncon} which lacks all speaker information from e_S and W_{LoRA} .

The modified score \hat{s} is calculated by applying classifier-free guidance with the above unconditional scores as follows:

$$\hat{s}_{\theta^*}(X_t|c, e_S) = s_{\theta^*}(X_t|c, e_S) + \gamma_S \cdot (s_{\theta^*}(X_t|c, e_S) - s_{uncon}). \quad (5)$$

Here, γ_S represents the gradient scale which determines the intensity of the additional speaker information.

We perform TTS with the 4 methods (adjusting α and 3 candidates for s_{uncon}) described above and observe that methods other than applying classifier-free guidance with $s_{uncon} = s_{\theta^*}(X_t|c, e_\phi)$ lead to detrimental performance in speaker adaptation. Therefore, when generating samples with VoiceTailor, we adopt using $s_{uncon} = s_{\theta^*}(X_t|c, e_\phi)$ as our final method. The related results and analysis are presented in Section 3.2.2.

3. Experiments

3.1. Experimental Setup

3.1.1. Datasets

Similar to UnitSpeech, we train a multi-speaker diffusion-based TTS model using the LibriTTS dataset [23] which comprises 585 hours of speech-text data across 2,456 speakers. We employ the same speaker encoder as UnitSpeech trained on Vox-

Celeb 2 [24]. For evaluation purpose, we select 10 speakers from the LibriTTS `test-clean` subset choosing one reference audio for each speaker which is identical to the reference audio used in YourTTS [1]. We select 5 random samples for each speaker, resulting in a total of 50 samples for evaluation.

3.1.2. Training and Fine-tuning Details

For training the multi-speaker TTS model, we adhere to the UnitSpeech architecture but introduce a learnable unconditional speaker embedding e_ϕ during training to facilitate speaker classifier-free guidance. Training procedures are consistent with those of UnitSpeech. For speaker adaptation, we fine-tune W_{LoRA} for 500 iterations using the Adam optimizer [26] at a learning rate of 10^{-4} , which takes approximately 15 seconds using a single NVIDIA A100 GPU. Compared to UnitSpeech, VoiceTailor performs fine-tuning with a higher learning rate due to its significantly fewer parameters for adaptation. We set the LoRA rank r and scaling factor α to 16 and 8, respectively. By setting $r = 16$, we fine-tune only 311K of the total 127M parameters of the model, which corresponds to 0.25% of the total and amounts to a size of 1.3 MB in storage.

3.1.3. Evaluation

During evaluation, we select UnitSpeech as our one-shot baseline. Additionally, we choose YourTTS [1] as the zero-shot TTS baseline which is trained on a similar scale of speech data, and XTTS *v2*, a powerful open-source zero-shot TTS model known to be trained on over 16,000 hours of data. For the vocoder, we use the official checkpoint of BigVGAN [25]. During sampling, we use the same LoRA scale $\alpha = 8$ as used in training, set the speaker gradient scale $\gamma_S = 1$, and use step size $\Delta t = 0.02$. All samples are resampled to 16kHz and are normalized to -27 dB for a fair comparison.

We utilize a test set of 50 sentences to evaluate the performance of VoiceTailor. We evaluate subjective audio quality and naturalness of generated samples through a 5-scale mean opinion score (MOS) and the speaker similarity with a 5-scale speaker similarity mean opinion score (SMOS). We also measure objective metrics with the speaker encoder cosine similarity (SECS), and the character error rate (CER) for evaluating pronunciation accuracy. The MOS and SMOS assessments are conducted using MTurk, while the SECS and CER measurements employ Resemblyzer package’s speaker encoder [27] and CTC-based Conformer [28], respectively. Following UnitSpeech, we generate each sentence 5 times for the SECS and CER measurements and average the values.

3.2. Results

3.2.1. Model Comparison

We conduct comparative evaluations of our model against various baselines in adaptive text-to-speech, with the results detailed in Table 1. As observed in Table 1, VoiceTailor is capable of synthesizing high-quality speech comparable or superior to the baselines, with accurate pronunciation accuracy.

From the SMOS results measuring speaker similarity, we find that VoiceTailor matches UnitSpeech and exhibits superior adaptation performance to YourTTS, a zero-shot approach utilizing similar amounts of data ($p < 0.01$ in the Wilcoxon signed-rank test). Despite using significantly less data, VoiceTailor outperforms XTTS *v2* in SMOS ($p < 0.05$), a zero-shot TTS model trained on vastly larger datasets with larger model size. Notably, fine-tuning only 0.25% of parameters results in

Table 1: Results of one/zero-shot adaptive TTS models including mean opinion score (MOS), character error rate (CER), and speaker similarity mean opinion score (SMOS) with 95% CI. The Amount of Dataset denotes the volume of data used to train the multi-speaker TTS model, measured in hours. # Params refers to the number of parameters utilized for fine-tuning / the total number of parameters.

Method	Amount of Dataset	Fine-tuning	# Params	5-scale MOS	CER (%)	5-scale SMOS
Ground Truth	-	-	-	4.36 \pm 0.05	0.70	4.38 \pm 0.06
Mel + BigVGAN [25]	-	-	-	4.23 \pm 0.07	0.73	4.16 \pm 0.08
VoiceTailor	\approx 585 hrs	✓	0.311M / 127M	4.19 \pm 0.07	1.33	4.06 \pm 0.09
UnitSpeech [17]	\approx 585 hrs	✓	119M / 127M	4.13 \pm 0.07	1.24	4.08 \pm 0.10
XTTS v2	> 16,000 hrs	✗	0 / 467M	4.14 \pm 0.08	1.18	3.85 \pm 0.11
YourTTS [1]	\approx 474 hrs	✗	0 / 87M	3.87 \pm 0.09	2.78	3.67 \pm 0.10

Table 2: CER and SECS results for design choices. The final setup marked in **bold**. “attn + others”: injection of adapters to all linear layers in addition to the attention modules.

		CER (%)	SECS
LoRA Modules	attn	1.33	0.942
	attn + others	1.39	0.941
LoRA Rank r (# Trainable Params)	2 (39K)	1.37	0.939
	4 (78K)	1.41	0.939
	8 (156K)	1.47	0.940
	16 (311K)	1.33	0.942
	32 (623K)	1.35	0.941
LoRA Scale α	1	1.30	0.926
	2	1.24	0.937
	4	1.29	0.939
	8	1.33	0.942
LR / # Iters	$2 \cdot 10^{-5} / 500$	1.23	0.912
	$2 \cdot 10^{-5} / 2000$	1.25	0.942
	$10^{-4} / 500$	1.33	0.942
	$10^{-4} / 2000$	1.49	0.942

comparable speaker similarity to UnitSpeech which fine-tunes the whole parameters, highlighting the efficiency over existing diffusion-based one-shot TTS models in the adaptation.

3.2.2. Analysis

We investigate the impact of various factors that could affect LoRA-based speaker adaptation. Results on design choices during the fine-tuning process are in Table 2, while results related to the speaker information strengthening methodology during inference are in Table 3.

Parameter-Efficient Fine-Tuning As in Table 2, additionally injecting trainable low-rank matrices into linear layers other than attention (*attn + others*) does not improve pronunciation accuracy and speaker similarity. This aligns with the observation in Section 2.2 that attention modules are crucial for speaker adaptation. Unlike UnitSpeech, which uses a learning rate of $2 \cdot 10^{-5}$, VoiceTailor requires a higher learning rate due to its adaptation with significantly fewer parameters. The choice of α for determining the scale of W_{LoRA} during fine-tuning indicates that comparable speaker similarities can be achieved as long as it is not defined as a small value (e.g., $\alpha = 1$). Even an extremely small LoRA rank ($r = 2$) degrades SECS slightly, suggesting that VoiceTailor can perform speaker adaptation with as few as 39K parameters (0.18 MB), should minor performance losses be deemed acceptable for significant parameter efficiency.

Speaker Information Strengthening Methods We explore various techniques to strengthen the speaker information in the sampling procedure. The quantitative results presented in Table 3 show that except for classifier-free guidance based on the

Table 3: CER and SECS results for speaker information strengthening techniques for sampling. The final setup marked in **bold**. “ $2.0 \cdot \alpha$ ”: doubles α used for training at inference.

		CER (%)	SECS
w/o strengthening	-	1.25	0.934
LoRA scale (sampling)	$2.0 \cdot \alpha$	7.46	0.863
Gradient scale γ_S ($s_{uncon} = s_{\theta^*}(X_t c, e_\phi)$)	1.0	1.33	0.942
	2.0	1.40	0.941
Gradient scale γ_S ($s_{uncon} = s_\theta(X_t c, e_S)$)	1.0	1.38	0.918
	2.0	1.40	0.895
Gradient scale γ_S ($s_{uncon} = s_\theta(X_t c, e_\phi)$)	1.0	1.26	0.929
	2.0	1.46	0.916

speaker embedding e_S ($s_{uncon} = s_{\theta^*}(X_t|c, e_\phi)$), other techniques deteriorate speaker adaptation performance. For example, elevating the LoRA scaling factor α above the value used for fine-tuning degrades both CER and SECS on a large scale. Thus, we only apply speaker embedding guidance with $\gamma_S = 1$.

4. Conclusion

We introduce VoiceTailor which is capable of performing high-quality personalized TTS with a pluggable and small personalized adapter. VoiceTailor maximizes parameter efficiency by careful injection of LoRA into pivotal modules for speaker adaptation based on the weight change ratio analysis, alongside exploring various guidance techniques to strengthen the speaker information. Consequently, we demonstrate that VoiceTailor is able to achieve performance comparable to fully fine-tuned adaptive TTS baselines with only 0.25% of the parameters and further show its robustness in real-world scenarios.

We believe that VoiceTailor will reduce the burden of building a personalized TTS system to support numerous new speakers efficiently. Nonetheless, there is room for further improvements in our parameter-efficient speaker adaptation. Future directions could include exploring methodologies for conducting speaker adaptation with even fewer parameters without performance degradation and extending the method to other adaptive speech synthesis tasks, such as any-to-any voice conversion.

5. Acknowledgements

This work was supported by Samsung Electronics (IO221213-04119-01), Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (RS-2021-II211343: AI Graduate School Program, SNU, RS-2022-II220959), National Research Foundation of Korea grant funded by MSIT (2022R1A3B1077720), and the BK21 FOUR program of the Education and Research Program for Future ICT Pioneers, SNU in 2024.

6. References

- [1] E. Casanova, J. Weber, C. D. Shulby, A. C. Junior, E. Gölge, and M. A. Ponti, “YourTTS: Towards zero-shot multi-speaker TTS and zero-shot voice conversion for everyone,” in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 17–23 Jul 2022, pp. 2709–2720. [Online]. Available: <https://proceedings.mlr.press/v162/casanova22a.html>
- [2] S. Kim, H. Kim, and S. Yoon, “Guided-tts 2: A diffusion model for high-quality adaptive text-to-speech with untranscribed data,” 2022.
- [3] M. Le, A. Vyas, B. Shi, B. Karrer, L. Sari, R. Moritz, M. Williamson, V. Manohar, Y. Adi, J. Mahadeokar, and W.-N. Hsu, “Voicebox: Text-guided multilingual universal speech generation at scale,” in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. [Online]. Available: <https://openreview.net/forum?id=gzCS252hCO>
- [4] C. Wang, S. Chen, Y. Wu, Z. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li, L. He, S. Zhao, and F. Wei, “Neural codec language models are zero-shot text to speech synthesizers,” 2023.
- [5] S. Kim, K. J. Shih, R. Badlani, J. F. Santos, E. Bakhturina, M. T. Desta, R. Valle, S. Yoon, and B. Catanzaro, “P-flow: A fast and data-efficient zero-shot TTS through speech prompting,” in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. [Online]. Available: <https://openreview.net/forum?id=zNA7u7wtIN>
- [6] K. Shen, Z. Ju, X. Tan, E. Liu, Y. Leng, L. He, T. Qin, sheng zhao, and J. Bian, “Naturalspeech 2: Latent diffusion models are natural and zero-shot speech and singing synthesizers,” in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=Rc7dAwVL3v>
- [7] Z. Zhang, Q. Tian, H. Lu, L. Chen, and S. Liu, “Adadurian: Few-shot adaptation for neural text-to-speech with durian,” *CoRR*, vol. abs/2005.05642, 2020. [Online]. Available: <https://arxiv.org/abs/2005.05642>
- [8] H. B. Moss, V. Aggarwal, N. Prateek, J. I. González, and R. Barra-Chicote, “Boffin tts: Few-shot speaker adaptation by bayesian optimization,” *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7639–7643, 2020. [Online]. Available: <https://api.semanticscholar.org/CorpusID:211044093>
- [9] C.-P. Hsieh, S. Ghosh, and B. Ginsburg, “Adapter-Based Extension of Multi-Speaker Text-To-Speech Model for New Speakers,” in *Proc. INTERSPEECH 2023*, 2023, pp. 3028–3032.
- [10] S. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, “Neural voice cloning with a few samples,” in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2018/file/4559912e7a94a9c32b09d894f2bc3c82-Paper.pdf
- [11] M. Chen, X. Tan, B. Li, Y. Liu, T. Qin, sheng zhao, and T.-Y. Liu, “Adaspeech: Adaptive text to speech for custom voice,” in *International Conference on Learning Representations*, 2021. [Online]. Available: <https://openreview.net/forum?id=Drynvt7gg4L>
- [12] Y. Yan, X. Tan, B. Li, T. Qin, S. Zhao, Y. Shen, and T.-Y. Liu, “Adaspeech 2: Adaptive text to speech with untranscribed data,” in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6613–6617.
- [13] E. J. Hu, yelong shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, “LoRA: Low-rank adaptation of large language models,” in *International Conference on Learning Representations*, 2022. [Online]. Available: <https://openreview.net/forum?id=nZeVKeeFY9>
- [14] X. L. Li and P. Liang, “Prefix-tuning: Optimizing continuous prompts for generation,” in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds. Online: Association for Computational Linguistics, Aug. 2021, pp. 4582–4597. [Online]. Available: <https://aclanthology.org/2021.acl-long.353>
- [15] J. Ho, A. Jain, and P. Abbeel, “Denosing Diffusion Probabilistic Models,” in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Curran Associates, Inc., 2020, vol. 33.
- [16] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman, “Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 22 500–22 510.
- [17] H. Kim, S. Kim, J. Yeom, and S. Yoon, “UnitSpeech: Speaker-adaptive Speech Synthesis with Untranscribed Data,” in *Proc. INTERSPEECH 2023*, 2023, pp. 3038–3042.
- [18] N. Kumari, B. Zhang, R. Zhang, E. Shechtman, and J.-Y. Zhu, “Multi-concept customization of text-to-image diffusion,” in *CVPR*, 2023.
- [19] Y. Li, R. Zhang, J. Lu, and E. Shechtman, “Few-shot image generation with elastic weight consolidation,” in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, ser. NIPS’20. Red Hook, NY, USA: Curran Associates Inc., 2020.
- [20] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. Kudinov, “Grad-TTS: A Diffusion Probabilistic Model for Text-to-Speech,” in *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, ser. Proceedings of Machine Learning Research, vol. 139. PMLR, 2021, pp. 8599–8608.
- [21] 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 *Proc. Interspeech 2021*, 2021.
- [22] J. Ho and T. Salimans, “Classifier-free diffusion guidance,” in *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. [Online]. Available: <https://openreview.net/forum?id=qw8AKxfYBI>
- [23] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, “LibriTTS: A Corpus Derived from LibriSpeech for Text-to-Speech,” in *Proc. Interspeech 2019*, 2019, pp. 1526–1530.
- [24] J. S. Chung, A. Nagrani, and A. Zisserman, “Voxceleb2: Deep speaker recognition,” in *INTERSPEECH*, 2018.
- [25] S. gil Lee, W. Ping, B. Ginsburg, B. Catanzaro, and S. Yoon, “BigVGAN: A universal neural vocoder with large-scale training,” in *The Eleventh International Conference on Learning Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=iTtGCMDEzS_
- [26] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: <http://arxiv.org/abs/1412.6980>
- [27] G. Louppe, “Resemblyzer,” <https://github.com/resemble-ai/Resemblyzer>, 2019.
- [28] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, “Conformer: Convolution-augmented Transformer for Speech Recognition,” in *Proc. Interspeech 2020*, 2020, pp. 5036–5040.