Diff Transferer: Any speaker adaptive Text-to-Speech with diffusion

Jun Wang, Dawei Li, Jinhe Wen, Fei Teng
University of California San Diego
La Jolla 92093, USA
{juw048,dal034,jhw,feteng}@ucsd.edu

1 Introduction

Text-to-speech (TTS) systems, built to generate a natural and emotional voice from a given text, have received lots of attention in recent years due to the great advances of deep generative models [39]. However, most TTS systems were tailored from a single-speaker voice. There is still significant interest in any-speaker adaptive TTS within few-/zero-shot scenarios because of the high price of gathering customer voice data in the real world. In this work, we want to build a model to mimic target speakers' styles with a few reference voices of a specific speaker, which are unseen during training.

Most methods in style transfer TTS focus on extracting the style vector from the reference voice and combining it with the phoneme embedding to generate styled mel-spectrograms, which would be input into the vocoder later [36, 43, 16]. To solve the few-/zero-shot problem, meta-learning is also applied in some works [28, 14]. But the generated voice quality of the meta-learning model is still at a low level.

Recently, due to the huge progress made by diffusion methods in image generation, there are also some works that proposed diffusion-based methods in any-speaker adaptive TTS [17, 18]. Although they outperform some previous methods, here we argue that those diffusion-based methods in any-speaker adaptive TTS still get some limitations. Firstly, they ignore **modal gap** in TTS. Unlike image generation which keeps the modal unchanged in input and output, in TTS, the model aims to generate corresponding speech according to the given text. Intuitively, that process involves modal transfer from text to speech and needs more detailed mechanism designs to serve as guidance. However, current works ignore that difference and apply the diffusion directly to the existing TTS architecture. Additionally, all the current diffusion-based TTS methods first sample white Gaussian noise and use the text and reference speech information to guide the restoration. That process always needs a large step size thus causing efficiency problems in both training and inference.

In this work, we propose Diff Transferer and leverage the recent advances in shallow diffusion mechanism [24] and controllable diffusion process [21] to solve the above problems. As Figure 1 shows, With the shallow diffusion mechanism applied, our model can transfer the speech generated by the TTS backbone into the target speaker's voice in several steps. Also, we propose two novel losses in the diffusion process to fuse the text and speech information into mel-spectrograms more smoothly.

Specifically, we want to use a style discriminator and a content discriminator to start generation at a shallow step smaller than the total number of diffusion steps. In this way, we can find a stage in which the style is masked while maintaining the content unchanged, which means decoupling style and text. Then we reverse the process to get a route from one speaker to another, similar to a diffusion GAN model [44].

Additionally, since an evaluation for cloning one's voice is hard to get, using the cosine distance between two speaker embeddings as the index of reference [17] would largely rely on the performance of the extracting model. The better your speaker embedding, the better evaluation or loss function you

can get. So in this work, we follow the previous works [28] to train our generator and discriminator simultaneously, which is the idea of GAN.

So given a text and some reference voices, we can first use FastSpeech 2 [32] as our base TTS model architecture to generate a mel-spectrogram with the input text as our original speaker. Then based on the reference voice, our Diff-Style Transferer would adapt its parameter in the diffusion model and transfer the original voice to the target one.

2 Literature Survey

2.1 Diffusion Model

In TTS tasks, deep generative models, e.g., GAN [6, 10, 11], VAE [34, 19, 8], normalizing flow [7, 33, 20], and diffusion models [35, 13, 26], have been widely used and shown great potential. Here we mainly introduce the theory of diffusion probabilistic model [37, 12]. With the probabilistic diffusion model, we can gradually convert the raw data into Gaussian distribution by adding noise and then learns the reverse process to restore the data from Gaussian white noise or some intermediate states. For the forward and backward process, [38] proposes a unified framework that further extends the capabilities of score-based generative models through stochastic differential equations (SDEs) and ordinary differential equations (ODEs). Based on the idea of mapping a uniform distribution on a high-dimensional hemisphere into any data distribution, [45] leverages the concept of the electromagnetic field in physics and proposes a new "Poisson flow" generative model.

There are some works that apply the diffusion model to post-process the generated mel-spectrograms in the style transfer TTS tasks. [17, 18] However, as we mentioned in Section 1, all of those methods get limitations due to the modal gap and efficiency problems. To solve those problems, we propose Diff Transferer, a shallow diffusion-based style transfer TTS model with a controllable diffusion process.

2.2 Few-/Zero-shot Style Transfer TTS

In few-shot and zero-shot TTS tasks, a popular and effective approach is fine-tuning the model with a few audio samples of the target speaker based on a pre-trained model [4, 1, 3]. However, audio samples and corresponding transcripts of the target speakers are hard and expensive to gather. Additionally, hundreds of fine-tuning steps are required in that method, limiting its applicability to real-world scenarios.

Another approach is to extract a latent vector from given reference speech audio that represents the style of this audio, such as speaker identity, prosody, and emotion [36, 43, 16]. But these methods show low adaptation performance on new speaker problems since they heavily rely on the diversity of the speakers in the training dataset.

Meta-learning [40], or learning to learn, has been widely used in a series of few-/zero-shot tasks and achieved promising performance. There are also some works that try to adopt Meta-Learning in TTS. [29] introduce an approach in multi-lingual speech synthesis with meta-learning and improve the quality of the synthesized sound successfully even with less training data. [25] apply model-agnostic meta-learning (MAML) algorithm [9] in emotional style transfer to pre-trained a TTS model, which is highly sensitive to the few-sample adaptation. Following them, [14] also use MAML in the multi-speaker adaptation and achieve great performance with few training samples. Also, [28] propose a new TTS model with meta-learning that could synthesize high-quality voice and adapt to the new speakers smoothly. Besides, [17] combines the diffusion model with meta-learning to deal with the few-shot problem.

In this work, we follow previous works and adopt the diffusion method in any speaker adaptive TTS and make adjustments at both model architecture and loss to make the diffusion fit better into the TTS task.

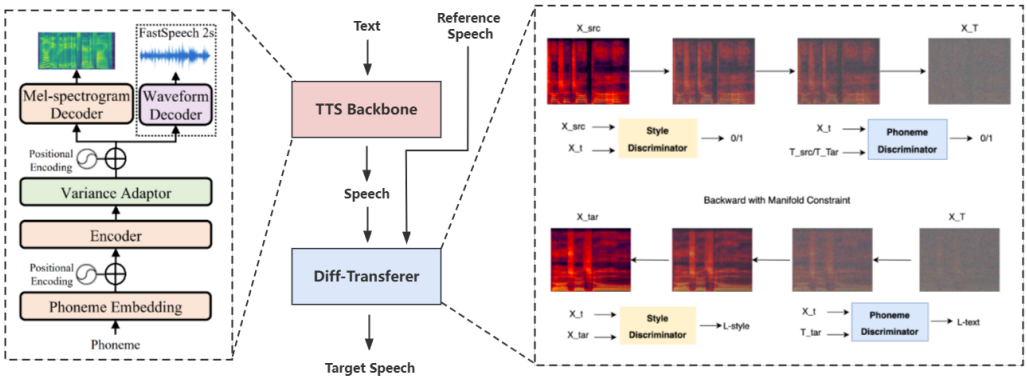


Figure 1: Overview architecture of our proposed Diff-Transferer

3 Proposed Method

3.1 DDPM Sampling

In the forward process of DDPMs [12] $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$, we continually add Gaussian noise to a clean speech $\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0)$ at every time step t:

$$q(\boldsymbol{x}_T|\boldsymbol{x}_0) := \prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}), \quad where \quad q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) := N(\boldsymbol{x}_t; \sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I}) \tag{1}$$

Where $\{\beta\}_{t=0}^{T}$ is a variance schedule to control the ratio of the original content and added noise. In practice, we rewrite the formulation and make the forward process's sampling in one step:

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \quad where \quad \epsilon \sim N(\boldsymbol{0}, \boldsymbol{I}) \tag{2}$$

Here $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_t$. It should be noticed that the reversed step of the forward process $q(x_t - 1|x_t)$ is intractable, so DDPM learns to maximize its variational lower bound $p_\theta(x_{t-1}|x_t)$ alternatively. Thus the reverse process can be formulated as another Markov chain with learned mean and fixed variance:

$$p_\theta(\boldsymbol{x}_{0:T}) := p_\theta(\boldsymbol{x}_T) \prod_{t=1}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t), \quad where \quad p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) := N(\boldsymbol{x}_{t-1}; \boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \sigma_t^2 \boldsymbol{I}) \tag{3}$$

and

$$\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t) := \frac{1}{\sqrt{\alpha_t}}(\boldsymbol{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)) \tag{4}$$

We can get the $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t))$ by optimizing the below objective:

$$\min_\theta L(\theta), \quad where \quad L(\theta) := \mathbb{E}_{t, \boldsymbol{x}_0, \boldsymbol{\epsilon}}[||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon})||^2] \tag{5}$$

After Training, we can conduct a reverse diffusion process and sample from $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ using:

$$\boldsymbol{x}_{t-1} = \boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t) + \sigma_t \boldsymbol{\epsilon} = \frac{1}{\sqrt{\alpha_t}}(\boldsymbol{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)) + \sigma_t \boldsymbol{\epsilon} \tag{6}$$

3.2 Conditional diffusion with Manifold Constraints

In the DDPM sampling process, we have a discrete form, as shown below

$$\boldsymbol{x}_i = a_i \boldsymbol{x}_0 + b_i \boldsymbol{z}, \quad \boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{I}) \tag{7}$$

$$\boldsymbol{x}_{i-1} = \boldsymbol{f}\left(\boldsymbol{x}_i, \boldsymbol{s}_{\theta^*}\right) + g\left(\boldsymbol{x}_i\right) \boldsymbol{z}, \quad \boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{I}) \tag{8}$$

where f(x,t) represents drift coefficient, g(t) represents diffusion coefficient, and S_{θ^*} is the trained score function.

And we want to apply a projection measurement y to retrieve the unknown x:

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{\epsilon}, \quad \boldsymbol{y} \in \mathbb{R}^m, \boldsymbol{H} \in \mathbb{R}^{m \times n} \tag{9}$$

Following equation 7,8, we can rewrite our Projection-based measurement constraint sample process in this form:

$$\boldsymbol{x}'_{i-1} = \boldsymbol{f}\left(\boldsymbol{x}_i, s_\theta\right) + g\left(\boldsymbol{x}_i\right) \boldsymbol{z}, \quad \boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{I}) \tag{10}$$

$$\boldsymbol{x}_{i-1} = \boldsymbol{A}\boldsymbol{x}'_{i-1} + \boldsymbol{b}_i, \text{ where } \boldsymbol{A}, \boldsymbol{b}_i \text{ are functions of } \boldsymbol{H}, \boldsymbol{y}, \text{ and } \boldsymbol{x}_0 \tag{11}$$

With Bayes Rule $\nabla_{\boldsymbol{x}} \log p(\boldsymbol{x} \mid \boldsymbol{y}) = \nabla_{\boldsymbol{x}} \log p(\boldsymbol{x}) + \nabla_{\boldsymbol{x}} \log p(\boldsymbol{y} \mid \boldsymbol{x})$ and Tweedie's formula, after replacing the equation 10 with a new score function, we can get:

$$\boldsymbol{x}'_{i-1} = \boldsymbol{f}\left(\boldsymbol{x}_i, \boldsymbol{s}_\theta\right) - \alpha \frac{\partial}{\partial \boldsymbol{x}_i} \left\|\boldsymbol{W}\left(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}_i\right)\right\|_2^2 + g\left(\boldsymbol{x}_i\right) \boldsymbol{z}, \quad \boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{I}) \tag{12}$$

From equation 12, we can add conditional loss to guide the diffusion sample process to do the style transfer, as shown in equations 16-19.

3.3 Shallow Diffusion Mechanism

Inspired by [24], we can leverage fastspeech2 to generate a speech with given target text and follow the forward process in diffusion until the mel-spectrogram is style-agnostic. In this way, we can make better use of the prior knowledge and reach a state where the style of the mel-spectrogram is mostly disentangled and ready to add the target style. As shown in Figure 2, we use a style discriminator to extract the style embedding and compute their cos-distance to judge whether the style is the same as the source input at the t step. We stop the forward process once the state is reached to shallow the diffusion. Since the generating voice from fastspeech2 would probably have the same style, the shallow t-steps could be set as a hyperparameter. Here, a phoneme discriminator is also used if we input a different text instead of the target one.

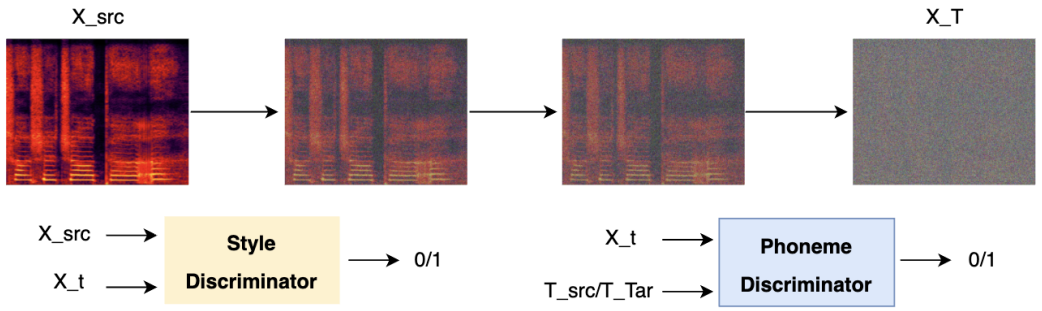


Figure 2: Forward Process until Style-agnostic

3.4 Style Loss

CLIP Loss for style translation

The style discriminator predicts whether the speech follows the style of the target speaker. We follow [28] to design the style discriminator, comprising Spectral processing, Temporal processing, and Multi-head self-attention with a set of style prototypes $S = \{s_i\}_{i=1}^{K}$. Here, $s_i \in \mathbb{R}^N$ means the style prototype for the ith speaker, and K is the number of speakers in the training set. We input the style embedding $w_s \in \mathbb{R}^N$, and the style prototype is learned by the following classification loss (13).

$$\mathcal{L}_{cls} = -\log \frac{\exp\left(w_s^T s_i\right))}{\sum_{i'} \exp\left(w_s^T s_{i'}\right)} \tag{13}$$

With the trained Style discriminator D_s, we can extract the style embedding from the mel-spectrograms and compare their similarity with cos-distances in (14).

$$\ell_{clip}\left(\boldsymbol{X}_{tar}, \boldsymbol{X}_t\right) := -\operatorname{sim}\left(D_s\left(\boldsymbol{X}_{tar}\right), D_s(\boldsymbol{X}_t)\right) \tag{14}$$

Content Loss for style translation

Since the two inputs here are different data types, we design a multi-modal phoneme discriminator with contrast learning to find a connection between the phoneme and mel-spectrograms. In (15), E_p E_m are the encoders of phoneme and mel-spectrograms.

$$\ell_{text}\left(\boldsymbol{X}_{tar}, \boldsymbol{X}_t\right) := -\operatorname{sim}\left(E_p\left(\boldsymbol{T}_{tar}\right), E_m(\boldsymbol{X}_t)\right) \tag{15}$$

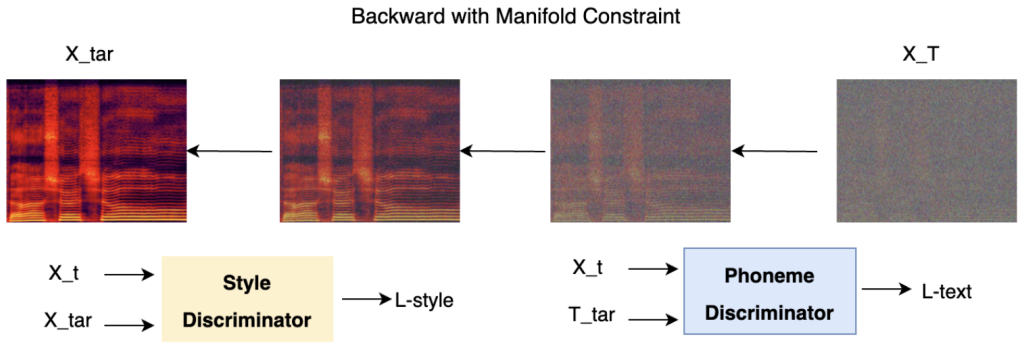


Figure 3: Backward with Manifold Constraint

3.5 Total Loss

The final loss for the transfer reverse diffusion is given by the following equation.

$$\ell_{\text{total}} = \lambda_1 \ell_{clip} + \lambda_2 \ell_{text} \tag{16}$$

In the backward process, we leverage the total loss as in [21] to sample:

$$\boldsymbol{x}'_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t, t\right)\right) + \sigma_t \boldsymbol{\epsilon} \tag{17}$$

$$\boldsymbol{x}_{t-1} = \boldsymbol{x}'_{t-1} - \nabla_{\boldsymbol{x}_t} \ell_{total}\left(\hat{\boldsymbol{x}}_0\left(\boldsymbol{x}_t\right)\right) \tag{18}$$

where $\hat{\boldsymbol{x}}_0\left(\boldsymbol{x}_t\right)$ refers to the estimated clean image from the sample using the Tweedie's formula [5].

$$\hat{\boldsymbol{x}}_0\left(\boldsymbol{x}_t\right) := \frac{\boldsymbol{x}_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t, t\right) \tag{19}$$

4 Experiments Setting

4.1 preliminary results

We have reproduced the code in Fastspeech2, Diffsinger, and the Meta-Stylespeech.

4.2 Datasets/Corpus

- **LibriTTS**[46]: a multi-speaker English corpus, which is derived from the audio and text materials of the LibriSpeech[30] corpus. The corpus of LibriTTS consists of 585 hours of English speech data at a 24kHz sampling rate from 2,456 speakers and their corresponding texts.
- **VCTK**[41]: includes speech data uttered by 110 English speakers with various accents. Each speaker reads out about 400 sentences, which were selected from a newspaper, the rainbow passage, and an elicitation paragraph used for the speech accent archive. All recordings were converted into 16 bits, were downsampled to 48 kHz, and were manually end-pointed.
- **LJSpeech**[15]: a public domain speech dataset consisting of 13,100 short audio clips of a single speaker reading passages from 7 non-fiction books. A transcription is provided for each clip. Clips vary in length from 1 to 10 seconds and have a total length of approximately 24 hours.
- **ESD**[47]: an Emotional Speech Database for voice conversion research. The ESD database consists of 350 parallel utterances spoken by 10 native English and 10 native Chinese speakers and covers 5 emotion categories (neutral, happy, angry, sad and surprise). More than 29 hours of speech data were recorded in a controlled acoustic environment. The database is suitable for multi-speaker and cross-lingual emotional voice conversion studies.
- **Opencpop**[42]: a publicly available Chinese Mandarin singing corpus designed for singing voice synthesis. The corpus consists of 100 popular Mandarin songs performed by a female professional singer. Audio files are recorded with studio quality at a sampling rate of 44,100 Hz and the corresponding lyrics and musical scores are provided. All singing recordings have been phonetically annotated with phoneme boundaries and syllable (note) boundaries.

4.3 Data Preprocessing

To begin with, we split the dataset into a training set and a validation set correspondingly. Text sequences are converted into phoneme sequences [23], which serve as input. Then, an audio is downsampled to 16kHz, the leading and trailing silence are also trimmed with Librosa[27]. A spectrogram is extracted with a FFT size of 1024, hop size of 256, and windowsize of 1024. Then, the spectrogram is converted to a mel-spectrogram with 80 frequency bins. Additionally, ground-truth pitch and energy are averaged by duration, to phoneme-level ones.

4.4 Baselines

Below are some of the baseline models for comparison.

- **GT (oracle)**: a ground truth speech.
- **GTmel (oracle)**: a speech synthesized by MelGAN vocoder using Ground-Truth mel-spectrogram.
- **StyleSpeech**[28] a model which generates multi-speaker speech from a single speech audio with the style-adaptive layer normalization and the mel-style encoder.
- **Meta-StyleSpeech**[28]: a zero-shot any-speaker adaptive model based on the meta-learning, which is an extension of StyleSpeech, with two additional discriminators to guide the generator.

- **YourTTS**[2]: a zero-shot any-speaker adaptive model based on the flow-based model[18].
- **Grad-TTS**[31]: a model that modifies the Grad-TTS into the any-speaker version.
- **Styler**[22]: a non-autoregressive TTS framework with style factor modeling that achieves rapidity, robustness, expressivity, and controllability at the same time.

4.5 Evaluation

Following previous works, we choose both automatic and manual evaluation metrics to prove the effectiveness of our model:

- **Speaker Embedding Cosine Similarity (SECS)**: a indication of the similarity between two speaker embeddings.
- **Mel-cepstral Distortion (MCD)**: the compatibility between the spectra of two audio sequences.

4.6 Evaluation on Trained Speakers

In this experiment, we test our model by directly using the samples in the testing set corresponding to the dataset we use to train our model. The speakers of those samples have been seen by the model during the training stage.

4.7 Unseen Speaker Adaptation

In this experiment, we aim to evaluate our model's transfer ability to unseen speakers. Specifically, we use another dataset within which the speaker of each sample is unavailable to the model during the training stage. By doing so, we hope to test our model's robustness and generalizing ability in the zero-shot scenario.

For both evaluations, we compare our model with baseline models in the following aspects:

- The quality of speech (using MOS).
- The similarity between synthesized speech for seen speakers and the reference speech (using SECS, CER, SMOS, and MCD).

5 Results and Conclusion

5.1 Hyperparameters

We sampled the LibriTTS dataset tracks at 22.05 kHz for the hyperparameters of audio signal processing. Each audio input is a 4 to 14 seconds audio segment, each with one sentence randomly chosen from the training set. We applied STFT and mel-filter-bank to convert audio components into STFT spectrograms, then to mel-spectrograms. The window size is 1024, the hop size is 256, and the number of mel bins is 80. Figure 4 shows the pipeline's learning rate and phoneme, sentence, and word duration loss.

For hyperparameters of the training paradigm, we trained the model with 4 NVIDIA RTX 3090Ti GPUs with 16 batch size. We adopted the Adam optimizer with a 10^{-3} learning rate. The total time to train the singing voice synthesis model of 160 steps is around 24 hours until the model converges.

The loss function for converging the model is the same as our optimization target, the L1-Norm loss between the estimated mel-spectrogram and the ground truth mel-spectrogram.

Figure 5 shows the training mel loss on two training pipelines and the generative loss on Hifi-Gan Vocoder. From the loss, we can see the mel-spectrogram predict model fits well and contains most of the information as shown in figure 7. But the Hifigan model's loss is still high, which means the vocoder does not work very well, as shown in the demo. We also tried fine-tuning a pre-trained model on our LibriTTS dataset to get another vocoder. However, it still contains some noise in the generated voice, and we later would introduce some extra conditions to train a better vocoder.

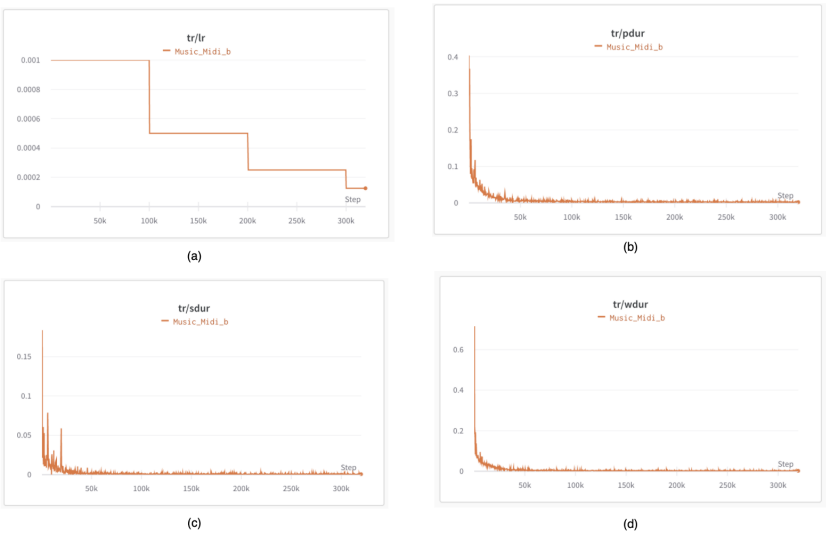


Figure 4: (a) The learning rate on training gradually declines. (b) The phoneme duration loss. (c) The sentence duration loss. (d) The word duration loss.

5.2 Training Results

From figure 7, the predicted f0 curve does not fit well in some minor parts. Thus, some details may be lost in the predicted mel-spectrogram. Besides, due to the target being to minimize the l1 loss in the model, the converging is not smooth to generate some details as the ground truth. Later multi-scale mel-spectrogram loss would be introduced to improve the performance. From figure6, we can see the style loss and phoneme loss during the training fluctuate within a specific range.

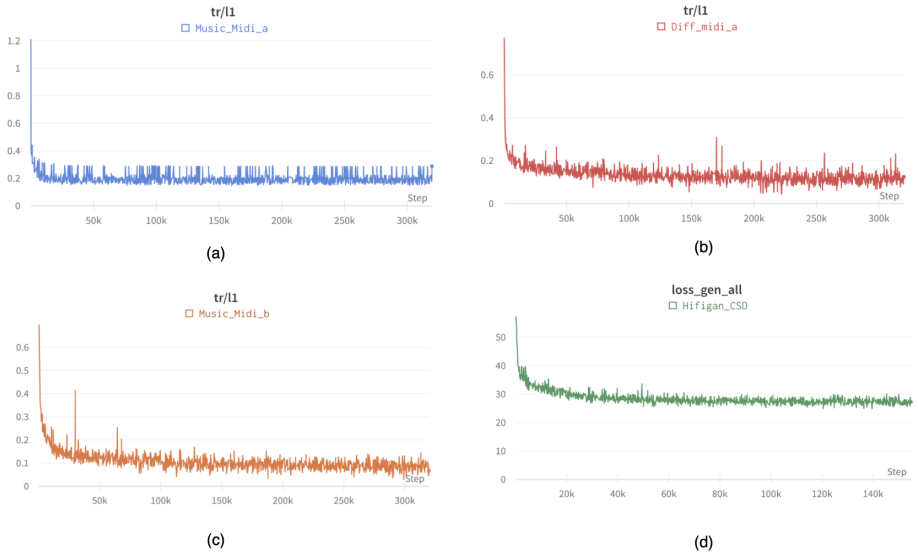
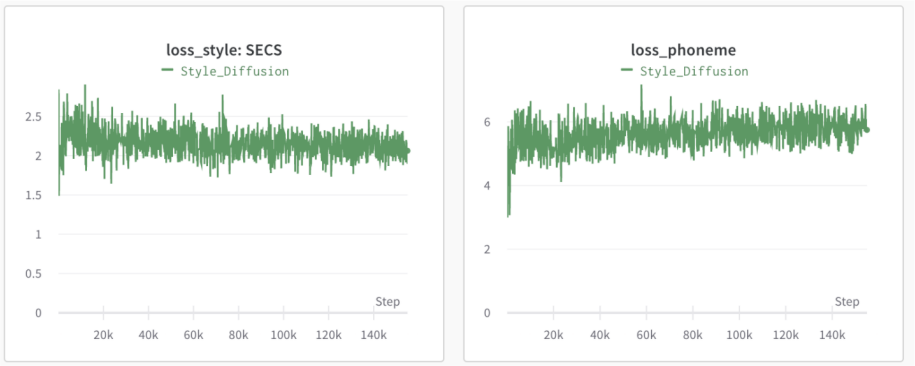


Figure 5: (a) The mel-loss on TTS model in the pipeline. (b) The mel-loss on shallow diffusion process in the pipeline. (c) The mel-loss on Diff-singer in the midi-b pipeline. (d) The total generative loss on Hifi-Gan Vocoder.

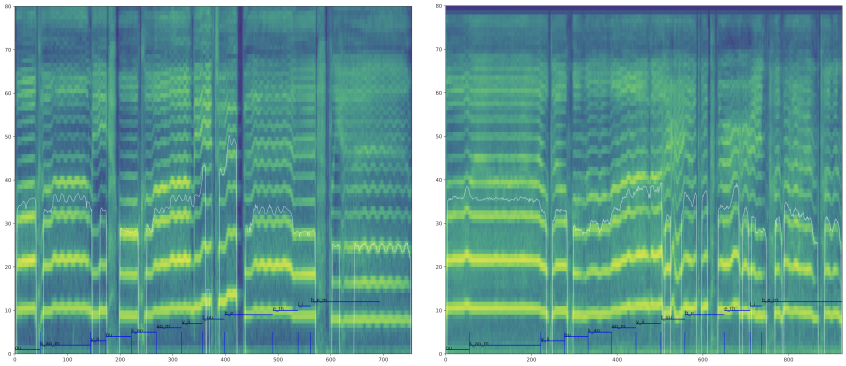


(a) (b)

Figure 6: (a) The style-loss on style discriminator. (b) The phoneme loss on phoneme discriminator.

5.3 Conclusion and Future Work

We have proposed a new pipeline for an any-speaker adaptive TTS model, which achieves a relatively good result in our final project. But it still requires a TTS backbone. The way to first generate a voice and then convert is not efficient. Both the training and inference are still slow. Maybe we can combine the transfer process in the middle of any speaker adaptive TTS in the future. Besides, the evaluation is still lost. Although compared with any adaptive speaker, it seems reasonable, the quality and the copy are still not high. Maybe we can evaluate the speaker style similarity between two voices better instead of comparing the cos distance in pertained Speaker Verification models. Moreover, we can leverage meta-learning to improve our models, transfer emotion, or even sing voice or techniques later.



(a) ground-truth (b) predict output

Figure 7: The compare for ground-truth and predict mel spectrogram of the sentence "oh come ye oh come ye to Bethlehem" in midi-a pipeline, including f0 curve (white line) and duration curve (phoneme with blue line).

Appendix

References

[1] Samy Bengio, Hanna M Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett. Advances in neural information processing systems 31: Annual conference on neural information processing systems 2018, 2018.

[2] Edresson Casanova, Julian Weber, Christopher D Shulby, Arnaldo Candido Junior, Eren Gölge, and Moacir A Ponti. Yourtts: Towards zero-shot multi-speaker tts and zero-shot voice conversion

for everyone. In *International Conference on Machine Learning*, pages 2709–2720. PMLR, 2022.

[3] Mingjian Chen, Xu Tan, Bohan Li, Yanqing Liu, Tao Qin, Sheng Zhao, and Tie-Yan Liu. Adaspeech: Adaptive text to speech for custom voice. *arXiv preprint arXiv:2103.00993*, 2021.

[4] Yutian Chen, Yannis Assael, Brendan Shillingford, David Budden, Scott Reed, Heiga Zen, Quan Wang, Luis C Cobo, Andrew Trask, Ben Laurie, et al. Sample efficient adaptive text-to-speech. *arXiv preprint arXiv:1809.10460*, 2018.

[5] Corinna Cortes, N Lawarence, D Lee, M Sugiyama, and R Garnett. Advances in neural information processing systems 28. In *Proceedings of the 29th Annual Conference on Neural Information Processing Systems*, 2015.

[6] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1):53–65, 2018.

[7] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

[8] Carl Doersch. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*, 2016.

[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[11] Jie Gui, Zhenan Sun, Yonggang Wen, Dacheng Tao, and Jieping Ye. A review on generative adversarial networks: Algorithms, theory, and applications. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1, 2021.

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[13] Emiel Hoogeboom, Vıctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *International Conference on Machine Learning*, pages 8867–8887. PMLR, 2022.

[14] Sung-Feng Huang, Chyi-Jiunn Lin, Da-Rong Liu, Yi-Chen Chen, and Hung-yi Lee. Meta-tts: Meta-learning for few-shot speaker adaptive text-to-speech. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:1558–1571, 2022.

[15] Keith Ito and Linda Johnson. The lj speech dataset. `https://keithito.com/LJ-Speech-Dataset/`, 2017.

[16] Ye Jia, Yu Zhang, Ron Weiss, Quan Wang, Jonathan Shen, Fei Ren, Patrick Nguyen, Ruoming Pang, Ignacio Lopez Moreno, Yonghui Wu, et al. Transfer learning from speaker verification to multispeaker text-to-speech synthesis. *Advances in neural information processing systems*, 31, 2018.

[17] Minki Kang, Dongchan Min, and Sung Ju Hwang. Any-speaker adaptive text-to-speech synthesis with diffusion models. *arXiv preprint arXiv:2211.09383*, 2022.

[18] Jaehyeon Kim, Sungwon Kim, Jungil Kong, and Sungroh Yoon. Glow-tts: A generative flow for text-to-speech via monotonic alignment search. *Advances in Neural Information Processing Systems*, 33:8067–8077, 2020.

[19] Diederik P Kingma, Max Welling, et al. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392, 2019.

[20] Ivan Kobyzev, Simon JD Prince, and Marcus A Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE transactions on pattern analysis and machine intelligence*, 43(11):3964–3979, 2020.

[21] Gihyun Kwon and Jong Chul Ye. Diffusion-based image translation using disentangled style and content representation. *arXiv preprint arXiv:2209.15264*, 2022.

[22] Keon Lee, Kyumin Park, and Daeyoung Kim. STYLER: Style Factor Modeling with Rapidity and Robustness via Speech Decomposition for Expressive and Controllable Neural Text to Speech. In *Proc. Interspeech 2021*, pages 4643–4647, 2021.

[23] Younggun Lee, Suwon Shon, and Taesu Kim. Learning pronunciation from a foreign language in speech synthesis networks. *arXiv preprint arXiv:1811.09364*, 2018.

[24] Jinglin Liu, Chengxi Li, Yi Ren, Feiyang Chen, and Zhou Zhao. Diffsinger: Singing voice synthesis via shallow diffusion mechanism. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11020–11028, 2022.

[25] Songxiang Liu, Dan Su, and Dong Yu. Meta-voice: Fast few-shot style transfer for expressive voice cloning using meta learning. *arXiv preprint arXiv:2111.07218*, 2021.

[26] Shitong Luo, Yufeng Su, Xingang Peng, Sheng Wang, Jian Peng, and Jianzhu Ma. Antigen-specific antibody design and optimization with diffusion-based generative models. *bioRxiv*, pages 2022–07, 2022.

[27] Brian McFee, Colin Raffel, Dawen Liang, Daniel P Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8, pages 18–25, 2015.

[28] Dongchan Min, Dong Bok Lee, Eunho Yang, and Sung Ju Hwang. Meta-stylespeech: Multi-speaker adaptive text-to-speech generation. In *International Conference on Machine Learning*, pages 7748–7759. PMLR, 2021.

[29] Tomáš Nekvinda and Ondřej Dušek. One model, many languages: Meta-learning for multilingual text-to-speech. *arXiv preprint arXiv:2008.00768*, 2020.

[30] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

[31] Vadim Popov, Ivan Vovk, Vladimir Gogoryan, Tasnima Sadekova, and Mikhail Kudinov. Grad-tts: A diffusion probabilistic model for text-to-speech. In *International Conference on Machine Learning*, pages 8599–8608. PMLR, 2021.

[32] Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. Fastspeech 2: Fast and high-quality end-to-end text to speech. *arXiv preprint arXiv:2006.04558*, 2020.

[33] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR, 2015.

[34] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pages 1278–1286. PMLR, 2014.

[35] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10, 2022.

[36] RJ Skerry-Ryan, Eric Battenberg, Ying Xiao, Yuxuan Wang, Daisy Stanton, Joel Shor, Ron Weiss, Rob Clark, and Rif A Saurous. Towards end-to-end prosody transfer for expressive speech synthesis with tacotron. In *international conference on machine learning*, pages 4693–4702. PMLR, 2018.

[37] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[38] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[39] Xu Tan, Tao Qin, Frank Soong, and Tie-Yan Liu. A survey on neural speech synthesis. *arXiv preprint arXiv:2106.15561*, 2021.

[40] Sebastian Thrun and Lorien Pratt. Learning to learn: Introduction and overview. *Learning to learn*, pages 3–17, 1998.

[41] Christophe Veaux, Junichi Yamagishi, Kirsten MacDonald, et al. Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit. *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*, 2017.

[42] Yu Wang, Xinsheng Wang, Pengcheng Zhu, Jie Wu, Hanzhao Li, Heyang Xue, Yongmao Zhang, Lei Xie, and Mengxiao Bi. Opencpop: A high-quality open source chinese popular song corpus for singing voice synthesis. *arXiv preprint arXiv:2201.07429*, 2022.

[43] Yuxuan Wang, Daisy Stanton, Yu Zhang, RJ-Skerry Ryan, Eric Battenberg, Joel Shor, Ying Xiao, Ye Jia, Fei Ren, and Rif A Saurous. Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. In *International Conference on Machine Learning*, pages 5180–5189. PMLR, 2018.

[44] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion gans. *arXiv preprint arXiv:2112.07804*, 2021.

[45] Yilun Xu, Ziming Liu, Max Tegmark, and Tommi Jaakkola. Poisson flow generative models. *arXiv preprint arXiv:2209.11178*, 2022.

[46] Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. Libritts: A corpus derived from librispeech for text-to-speech. *arXiv preprint arXiv:1904.02882*, 2019.

[47] Kun Zhou, Berrak Sisman, Rui Liu, and Haizhou Li. Emotional voice conversion: Theory, databases and esd. *Speech Communication*, 137:1–18, 2022.