Interspeech 2024
1-5 September 2024, Kos, Greece

Bilingual and Code-switching TTS Enhanced with Denoising Diffusion Model and GAN

Huai-Zhe Yang, Chia-Ping Chen, Shan-Yun He, Cheng-Ruei Li

National Sun Yat-Sen University, Taiwan

yhc78887242@gmail.com,cpchen@cse.nsysu.edu.tw,vivian900611@gmail.com,
410821213@gms.ndhu.edu.tw

Abstract

In this paper, we propose a Mandarin-English bilingual and code-switching text-to-speech (TTS) system featuring a diffusion model and generative adversarial network (GAN) to improve the output speech. To address speaker consistency, we employ a feature separation architecture that converts language and speaker IDs into embeddings as input to the encoder. Subsequently, we employ two adversarial classifiers and two classifiers to separate language and speaker features. We integrate a modified diffusion model and discriminators to push for better speech quality and speaker consistency, especially for codeswtiching scenarios. On the MOS measure, the performance of the proposed TTS system differs only slightly from the ground truth data in monolingual speech and achieves MOS of 3.83 in the synthesis of code-switching speech.

Index Terms: text-to-speech, code-switching, generative adversarial network, diffusion probabilistic model

1. Introduction

Text-to-speech (TTS) systems are computer-based systems that convert input text into spoken language. These systems have been successfully applied in various fields such as speech-enabled customer service and voice translation. With the development of technology, the primary objective is to enhance the quality of synthesis and to achieve more fidelity of synthesized speech. To achieve this objective, various model architectures have been developed, including initial autoregressive (AR) models as well as non-autoregressive (non-AR) models. Subsequently, there has been further development of the model like generative adversarial network architectures (GAN) and denoising diffusion probabilistic models (DDPM).

In the AR TTS model, each frame is generated based on the previous frames, such as Tacotron2 [1], Transformer TTS [2], and Deep Voice 3 [3]. However, these models exhibit certain drawbacks, including slower synthesis speed and issues like word repetition or word skipping. To address these problems, various non-AR TTS models have been proposed, including FastSpeech2 [4] and Glow-TTS [5]. They utilize a parallel processing approach to solve the speed issue and incorporate alignment strategies during training to address issues such as missing words and repetitions. This architecture has been widely adopted in the field of Text-to-Speech. To improve the quality of generated speech, GAN-based models were later developed. In these architectures, there are typically two components: a generator and a discriminator. The generator produces results, while a discriminator assesses the quality of the output generated. This implies that the generator is trained not to minimize the distance to a specific result but rather to fool the discriminator. Numerous studies have demonstrated the effectiveness of this architecture in improving the quality of generated output, such as MelGAN [6] and GANSpeech [7]. There is another generative model, the DDPM, that has been extensively researched in recent years. It defines a Markov chain of diffusion steps to gradually introduce random noise to the data and subsequently learn to reverse the diffusion process, constructing the data samples from the noise. Compared to the GAN architecture, it provides enhanced stability and controllability, and generated results are almost identical, or even better. Consequently, several derivative models have been developed, including Prodiff [8], Diff-tts [9], and Grad-tts [10].

The previously mentioned architectures are primarily designed for monolingual systems. However, as an increasing number of individuals use English alongside their native languages for communication across various contexts, there is a growing interest in the development of bilingual systems. Bilingual systems use two languages within a single conversation or communication. Compared with the development of monolingual systems, the development of bilingual systems presents a more complex challenge. The major challenge is speaker inconsistency within synthesized bilingual sentences. In some studies [11, 12, 13], various solutions have been proposed. Nevertheless, many of these systems are mostly implemented on either AR or non-AR architectures.

To enhance the audio quality of a bilingual system, we initially refer to the architecture proposed in D. Xin et al. [11] and conduct experiments by implementing the system on the diffusion model framework. In D. Xin et al. [11], the primary method to address speaker inconsistency involves using two classifiers and two adversarial classifiers to separate speaker and language features. The language and speaker features are extracted by the extractor from the ground truth mel-spectrogram. We modify this architecture and incorporate it into the diffusion model as our initial bilingual system model. Within the diffusion model, we reference the Prodiff [8] architecture and adapt the encoder to Conformer [14]. To further enhance the synthesized speech, we reference Diff-GAN [15] and modify the JCU Discriminator to integrate it into our system. Finally, we modify the denoiser architecture in Prodiff by incorporating the SE-Block [16]. The SE-Block could provide weights to the noisy mel-spectrogram during denoising, allowing the model to more accurately assess the denoising effect at each stage. The final result demonstrates that each stage of our modification incrementally enhanced the performance of our bilingual system, from the initial implementation of the diffusion model to the incorporation of the JCU discriminator, and the modification of the denoiser architecture.

In the second section of this paper, we will introduce the model we modified and the total loss we used. In the third section, we will describe the dataset we used and our experiment setting. The fourth section will discuss the experimental results,

10.21437/Interspeech.2024-600

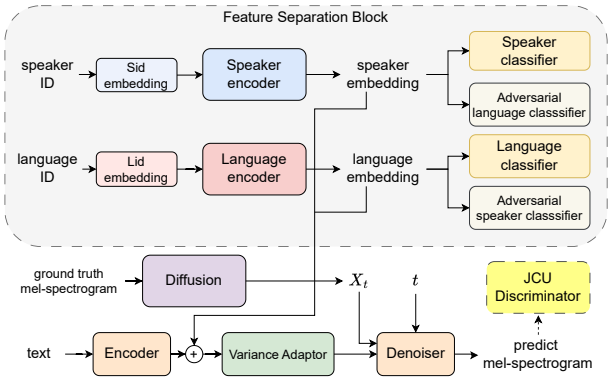

Figure 1: *The architecture of IFS-Diff and IFS-DiffGAN. The part without JCU Discriminator in IFS-Diff.* t *represents the diffusion time index.* X_t *represents the noisy spectrogram.*

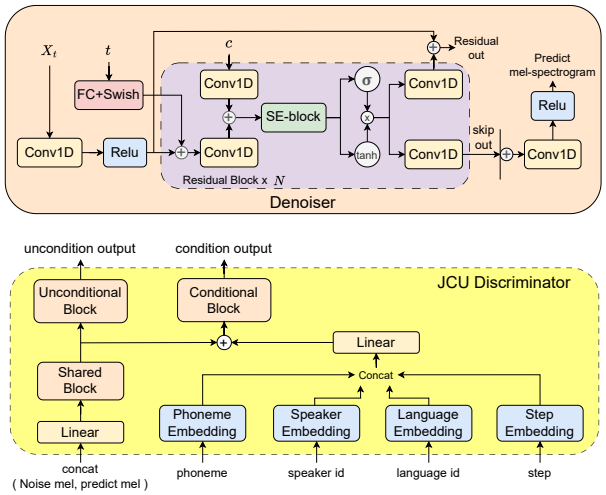

Figure 2: *Up: Our modified denoiser, Down: Our modified bilingual JCU Discriminator.* t *represents the diffusion time index.* X_t *represents the noisy spectrogram.* c *represents the output of the Variance Adaptor.*

including both objective and subjective evaluation.

2. Model

In this section, we introduce the architecture of our model. The first component is based on a modified approach inspired by D. Xin et al. [11] and Prodiff [8]. The second component discusses the JCU Discriminator [7] architecture, which we have modified and integrated into our architecture. For the third component, we enhance the model's performance by modifying the denoiser module in Prodiff. Lastly, we will discuss the total loss.

2.1. IFS-Diff

In bilingual TTS systems, the primary challenge is the speaker consistency in a code-switching sentence. Typically, when we train a bilingual system, the datasets from two languages are collected from different speakers. This may lead to different speaker voices within a single synthesized code-switching speech. The issue can be addressed by employing code-switching datasets or by utilizing datasets from the same speaker in a different language. However, obtaining code-switching datasets is a complex process and the quantity available is limited. Furthermore, publicly accessible datasets commonly lack cross-lingual data from the same speaker. To address this issue, we refer to the architecture proposed by D. Xin et al. [11], which aims to separate language features and speaker features. It utilizes the extractor to extract the features from the ground truth mel-spectrograms, followed by two encoders: a speaker encoder and a language encoder. Following the speaker encoder, there will be a speaker classifier and an adversarial language classifier. This architecture allows the speaker encoder's output embedding to contain speaker features while removing language features. Similar to the speaker encoder, the language encoder is followed by a language classifier and an adversarial speaker classifier. It allows the language encoder's output embedding to contain language features while removing speaker features. We refer to this architecture as the extractor feature separation (EFS) architecture. Although models based on this architecture can extract speaker features from unseen speakers' speech for synthesis, synthesizing speech from unseen speakers is not our primary object. We aim to focus on speakers within the training dataset and achieve high-quality synthesis speech.

In our modified EFS architecture, we remove the extractor and modify the input of the speaker encoder to be the speaker embedding converted by speaker ID, and the input of the language encoder to be the language embedding converted by language ID. We refer to this architecture as the ID feature separation (IFS) architecture. When we observe the language and speaker embedding using t-SNE [17], these embeddings from the same speaker in EFS architecture are often scattered around different domains. Conversely, utilizing lFS architecture allows these embeddings to be concentrated in similar domains. This indicates that the model can more accurately learn specific features. Later, we refer to the work by Gulati et al. [14] and find that Conformer architecture exhibits better performance compared to Transformer [18] architecture. Therefore, we make additional modifications to Prodiff by changing the encoder from Transformer to Conformer architecture. We combine the IFS framework with the modified Prodiff which we refer to this architecture as IFS-Diff. The architecture is shown in Fig. 1 and does not include the part of the JCU discriminator block. Our results show a better performance than the representative non-autoregressive model FastSpeech2. It indicates that our model contrasts with previous bilingual system research which uses primarily non-AR architectures, we have further enhanced overall performance by utilizing a diffusion model architecture.

2.2. IFS-DiffGAN

To improve synthesis quality, we refer to the method proposed by DiffGAN-TTS [15], which integrates DDPM with GAN. In comparison to utilizing only DDPM, this approach leads to higher fidelity and efficiency in speech synthesis. To integrate with DDPM and address the challenges posed by multispeaker TTS tasks, they made modifications to the JCU Discriminator proposed by GANSpeech [7]. Our system is more complex, as we not only deal with multispeaker issues but also language-related challenges. Therefore, we further modify the conditional input of the JCU Discriminator. We convert speaker ID and language ID into embeddings and add them to the conditional input. Furthermore, we introduce phoneme embeddings to constrain the Discriminator. The modified architecture of our JCU Discriminator is shown in Fig. 2.

In this architecture, training is conducted in two stages. During the first stage, the training process begins with the generator and is followed by the training of the discriminator. This iterative cycle is repeated continuously. We observe the training loss to determine when to stop training in this stage. In the second stage, the discriminator trained in the first stage is incorporated into the training process. However, during this stage, only the generator is trained.

2.3. IFS-DiffGAN-SE

In the diffusion model, the denoiser module is responsible for denoising tasks. Therefore, the performance of this module significantly influences the quality of our final generated results. In Prodiff, they followed Liu et al. [19], which adopt a non-causal Wavenet [20] architecture. In this architecture, there is a residual block that features a 1×1 convolution primarily designed for denoising the noisy mel-spectrogram. However, the denoising of the noisy mel-spectrogram may not always produce optimal results after each pass through the convolution. In that case, residual architecture consistently denoises subpar outcomes, leading to a decline in synthesis quality. Following this convolution, we incorporate a SE-block [16] as our denoiser module. By this approach, weights can be assigned to the noisy mel-spectrogram denoised at each step. This allows the model to determine which frequency dimension features in the noisy mel-spectrogram are more favorable during each denoising iteration, ultimately improving the denoiser's performance. We integrate this architecture with GAN architecture as IFS-DiffGAN-SE.

2.4. Total loss

In IFS-Diff, the loss function is defined as

$$\begin{aligned} L_{\text{tts}} = L_{\text{mel}} + L_{\text{pitch}} + L_{\text{duration}} + \\ L_{\text{energy}} + L_{\text{classifier}} + L_{\text{ssim}} \end{aligned} \tag{1}$$

where L_{mel} is using the Mean Absolute Error (MAE) as the loss metric between the predicted mel-spectrogram and the ground truth mel-spectrogram. L_{pitch}, L_{duration}, and L_{energy} represent the loss of the pitch predictor, duration predictor, and energy predictor, respectively. They are all computed using Mean Squared Error (MSE) as the loss metric. $L_{\text{classifier}}$ is the loss calculated by summing the losses of four classifiers using cross-entropy. L_{ssim} represents the loss associated with the structural similarity index between the predicted and ground truth mel-spectrograms.

In IFS-DiffGAN and IFS-DiffGAN-SE, the generator and discriminator are trained separately. There are two outputs in the discriminator, the conditional output (CO) and the unconditional output (UO). CO discriminates whether the input mel-spectrogram is generated data or original data, while UO discriminates whether it matches the conditional input. During the training of the generator, not only the components L_{tts} are considered, but also the generator loss. In contrast, during the training of the discriminator, only the discriminator loss is included. Both the discriminator and the generator are evaluated using the Mean Square Error (MSE) loss. The discriminator loss function is defined as

$$\begin{aligned} L_D = & \frac{1}{2}\left[\overbrace{D(\widehat{u})^2}^{UO} + \overbrace{D(\widehat{u},c)^2}^{CO}\right] + \\ & \frac{1}{2}\left[\overbrace{(D(u)-1)^2}^{UO} + \overbrace{(D(u,c)-1)^2}^{CO}\right] \end{aligned} \tag{2}$$

where L_D represents the discriminator's loss. $\widehat{u}$ represents the ground truth mel-spectrogram, while c denotes the conditional input. In our architecture, the conditional input includes phoneme, speaker ID, language ID, and diffusion step. u is the generated mel-spectrogram predicted by the generator. Since we attribute the ground truth data as label 1, the similarity is calculated by subtracting 1 during the computation. The generator loss function is defined as

$$L_G = \frac{1}{2}\left[\overbrace{(D(u)-1)^2}^{UO} + \overbrace{(D(u,c)-1)^2}^{CO}\right] \tag{3}$$

3. Experiment

In this section, we introduce the datasets employed in our model training. Subsequently, we'll review the details of the model architecture. Lastly, we discuss the training settings.

3.1. Dataset

We choose 65 Mandarin speakers from AISHELL-3 [21] and 106 English speakers from VCTK [22] for model training. Each Mandarin speaker contributes 300 to 480 utterances, totaling 26,747 utterances. Each English speaker contributes 250 to 450 utterances, totaling 39,720 utterances. Another 3,026 Mandarin utterances and 4,350 English utterances are used for the validation dataset. All audios are down-samples to 22kHz for model training.

3.2. Model Configurations

In the feature separation block, the speaker and language encoder consist of two fully connected layers. The adversarial classifier consists of a gradient reversal layer followed by two fully connected layers, while the language and speaker classifiers consist of a single fully connected layer. Input and hidden dimensions of the fully connected layer are 256, and output dimensions are based on a number of speakers or languages in the training set.

In Prodiff we modified, the embedding dimension is set to 256. The pitch and energy predictor employs a 5-layer 1D convolutional network, while the duration predictor utilizes a 7-layer architecture. The convolutional kernel size is set to 3. For the diffusion process, we use the cosine schedule and set β to 40. The diffusion step is 4. The residual block in the denoiser consists of 20 layers, and the final output is a predicted mel-spectrogram with 80 dimensions.

3.3. Training

We use the ESPnet toolkit [23] to train the TTS model in one NVIDIA V100 GPU. In IFS-Diff, we train for 150k steps and 3.5 Million batch-bins. The initial learning rate is 0.001. IFS-DiffGAN and IFS-DiffGAN-SE are trained in two stages. In the first stage, we train both the generator and the discriminator for a total of 100k steps and 3.5 Million batch-bins. In the second stage of training, only the generator is trained. The discriminator train in the first stage assists in updating the generator's parameters without training itself. The initial learning rate is set to 0.001 and the model is trained for 150k steps.

4. Results

Here, we will introduce our evaluation methods for the model's performance, which are divided into objective evaluation and

Table 1: *The objective comparison of results includes metrics such as MSE, MCD, and SSIM.*

Model	MSE (↓)	MCD (↓)	SSIM (↑)
IFS-FastSpeech2	0.518	11.02	0.535
EFS-Diff	0.303	11.16	0.525
IFS-Diff	0.263	10.92	0.567
IFS-DiffGAN	0.255	9.94	0.585
IFS-DiffGAN-SE	**0.247**	**9.75**	**0.594**

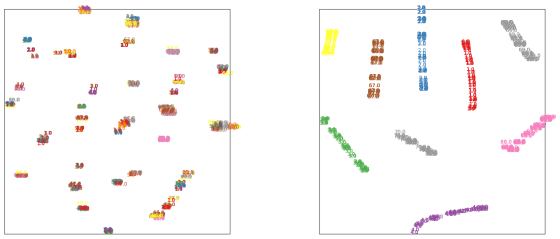

Figure 3: *Visualizing speaker embeddings using the t-SNE technique. Left: EFS-Diff, Right: IFS-Diff*

subjective evaluation. The comparison includes five models: IFS-Diff, IFS-DiffGAN, IFS-DiffGAN-SE, IFS architecture combined with FastSpeech2 (IFS-FastSpeech2), and EFS integrated with the diffusion model (EFS-Diff).

4.1. Objective Evaluation

We employ three objective standards: Mean Absolute Error (MAE), Mel-Cepstral Distortion (MCD) [24], and Structural Similarity Index (SSIM) [25]. To measure these standards, we utilize the mel-spectrogram predicted by the trained model and compare them with the ground truth mel-spectrogram. The evaluation is conducted on 200 test audio files, evenly divided between Mandarin and English. The results are presented in Table 1.

We can observe that IFS-Diff is better than the IFS-FastSpeech2 in the three metrics. It indicates that compared to previous bilingual system studies [11, 12, 13] which commonly employed the autoregressive (AR) or non-autoregressive (non-AR) architectures, modifying the system to the diffusion model architecture can improve performance. Additionally, IFS-Diff provides better performance than EFS-Diff in evaluation results. It indicates that our approach of converting ID to embeddings, while not capable of synthesizing speech from unseen speakers as proposed in the paper by D. Xin et al.[11], allows us to focus more on the quality of synthesizing speech from speakers in the training set. Figure 3 is the t-SNE [17] visualization of speaker embeddings for 9 randomly selected speakers from the dataset. For each speaker, 30 random speeches were chosen. The left side corresponds to the speaker embedding provided by EFS-Diff, while the right is provided by IFS-Diff. It can be observed that our approach effectively clusters embeddings of the same speaker in the same domain.

From Table 1, it can also be observed that incorporating the GAN structure into the IFS-Diff, resulting in IFS-DiffGAN, leads to improved performance across the three metrics. In the final architecture of IFS-DiffGAN-SE, we modify the denoiser structure from IFS-DiffGAN by incorporating the SE-Block. This addition to the denoiser enables the model to assign a weight during each denoising step, thereby enhancing the denoising performance. These evaluation metrics demonstrate that the modifications we made effectively enhance the model's performance.

Table 2: *MOS on ground-truth audio, IFS-FastSpeech2, EFS-Diff, IFS-Diff, IFS-DiffGAN, and IFS-DiffGAN-SE.*

	Mandarin	English	CS
ground-truth	4.51 ± 0.44	4.67 ± 0.38	-
IFS-FastSpeech2	2.86 ± 0.61	3.43 ± 0.77	2.62 ± 0.94
EFS-Diff	3.98 ± 0.46	4.01 ± 0.41	3.42 ± 0.56
IFS-Diff	4.05 ± 0.47	4.02 ± 0.66	3.47 ± 0.73
IFS-DiffGAN	4.15 ± 0.45	4.07 ± 0.44	3.48 ± 0.7
IFS-DiffGAN-SE	**4.27 ± 0.42**	**4.21 ± 0.48**	**3.83 ± 0.66**

4.2. Subjective Evaluation

To evaluate the synthesized audio files, we conduct Mean Opinion Score (MOS) tests in which ratings usually range from 1 to 5, with 1 representing poor quality and 5 representing excellent quality. The comparison results are presented in three types: Mandarin sentences, English sentences, and code-switching (CS) sentences. Each section includes ten sentences evaluated individually, with a total of twenty evaluators participating in the assessment. The results are presented in Table 2. Comparison between IFS-FastSpeech2, EFS-Diff, and IFS-Diff shows that the improvement in a diffusion model coupled with the modified FS architecture significantly improves the performance of the model. Then, we incorporate the diffusion model with GAN and modify the denoiser module. As shown in the result, the performance of IFS-DiffGAN-SE has the best performance in Mandarin, English, and CS synthesized utterances, especially where there is a significant gap in CS sentences. The result demonstrates that our model improvements enhance the performance of the bilingual system regardless of whether we combine the diffusion model with GAN or modify the Denoiser block.

5. Conclusions

In this paper, we initially transformed the conventional bilingual system which is based on non-autoregressive (non-AR) architecture into the diffusion model architecture. The results of the speech synthesized by the diffusion model architecture show higher quality compared to the traditional non-AR architecture. Following that, we combined the diffusion model with GAN architecture and then improved the denoising performance of the diffusion model's denoiser to achieve improved noise reduction efficiency. The comparative results also demonstrated that this approach has achieved certain improvements. We minimize the gap of quality between generated results and the ground truth speech. Although the synthesized code-switching speeches may not reach the exceptional level observed in monolingual sentences, the performance remains quite well. Our future work will be on further enhancing the quality of the synthesized code-switching speech. Through ongoing improvements in overall speech quality, our goal is to elevate the system, delivering a more realistic and natural speech synthesis experience.

6. 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 *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.

[2] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu, "Neural speech synthesis with transformer network," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 6706–6713.

[3] W. Ping, K. Peng, A. Gibiansky, S. O. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep voice 3: Scaling text-to-speech with convolutional sequence learning," *arXiv preprint arXiv:1710.07654*, 2017.

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

[5] J. Kim, S. Kim, J. Kong, and S. Yoon, "Glow-tts: A generative flow for text-to-speech via monotonic alignment search," *Advances in Neural Information Processing Systems*, vol. 33, pp. 8067–8077, 2020.

[6] K. Kumar, R. Kumar, T. De Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. De Brebisson, Y. Bengio, and A. C. Courville, "Melgan: Generative adversarial networks for conditional waveform synthesis," *Advances in neural information processing systems*, vol. 32, 2019.

[7] J. Yang, J.-S. Bae, T. Bak, Y. Kim, and H.-Y. Cho, "Ganspeech: Adversarial training for high-fidelity multi-speaker speech synthesis," *arXiv preprint arXiv:2106.15153*, 2021.

[8] R. Huang, Z. Zhao, H. Liu, J. Liu, C. Cui, and Y. Ren, "Prodiff: Progressive fast diffusion model for high-quality text-to-speech," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 2595–2605.

[9] M. Jeong, H. Kim, S. J. Cheon, B. J. Choi, and N. S. Kim, "Diff-tts: A denoising diffusion model for text-to-speech," *arXiv preprint arXiv:2104.01409*, 2021.

[10] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. Kudinov, "Grad-tts: A diffusion probabilistic model for text-to-speech," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8599–8608.

[11] D. Xin, T. Komatsu, S. Takamichi, and H. Saruwatari, "Disentangled speaker and language representations using mutual information minimization and domain adaptation for cross-lingual tts," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6608–6612.

[12] W.-N. Hsu, Y. Zhang, R. J. Weiss, Y.-A. Chung, Y. Wang, Y. Wu, and J. Glass, "Disentangling correlated speaker and noise for speech synthesis via data augmentation and adversarial factorization," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5901–5905.

[13] H. Ming, Y. Lu, Z. Zhang, and M. Dong, "A light-weight method of building an lstm-rnn-based bilingual tts system," in *2017 International Conference on Asian Language Processing (IALP)*. IEEE, 2017, pp. 201–205.

[14] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.

[15] S. Liu, D. Su, and D. Yu, "Diffgan-tts: High-fidelity and efficient text-to-speech with denoising diffusion gans," *arXiv preprint arXiv:2201.11972*, 2022.

[16] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[17] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[19] J. Liu, C. Li, Y. Ren, F. Chen, and Z. Zhao, "Diffsinger: Singing voice synthesis via shallow diffusion mechanism," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 36, no. 10, 2022, pp. 11 020–11 028.

[20] 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.

[21] Y. Shi, H. Bu, X. Xu, S. Zhang, and M. Li, "Aishell-3: A multi-speaker mandarin tts corpus and the baselines," *arXiv preprint arXiv:2010.11567*, 2020.

[22] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, "Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," 2016.

[23] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen *et al.*, "Espnet: End-to-end speech processing toolkit," *arXiv preprint arXiv:1804.00015*, 2018.

[24] 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. IEEE, 1993, pp. 125–128.

[25] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.