2025 IEEE INTERNATIONAL WORKSHOP ON MACHINE LEARNING FOR SIGNAL PROCESSING, AUG. 31– SEP. 3, 2025, ISTANBUL, TURKEY

UNSEEN SPEAKER AND LANGUAGE ADAPTATION FOR LIGHTWEIGHT TEXT-TO-SPEECH WITH ADAPTERS

Alessio Falai[⋆] *Ziyao Zhang*[⋆] *Akos Gangoly*[⋆]

[⋆] Amazon AGI
{falai,zhaziyao,gangakos}@amazon.com

ABSTRACT

In this paper we investigate cross-lingual Text-To-Speech (TTS) synthesis through the lens of adapters, in the context of lightweight TTS systems. In particular, we compare the tasks of unseen speaker and language adaptation with the goal of synthesising a target voice in a target language, in which the target voice has no recordings therein. Results from objective evaluations demonstrate the effectiveness of adapters in learning language-specific and speaker-specific information, allowing pre-trained models to learn unseen speaker identities or languages, while avoiding catastrophic forgetting of the original model's speaker or language information. Additionally, to measure how native the generated voices are in terms of accent, we propose and validate an objective metric inspired by mispronunciation detection techniques in second-language (L2) learners. The paper also provides insights into the impact of adapter placement, configuration and the number of speakers used.

Index Terms— text-to-speech, adapters, lightweight.

1. INTRODUCTION

The landscape of Artificial Intelligence (AI) has recently witnessed significant strides in Text-To-Speech (TTS) synthesis [1]. Contemporary TTS model architectures predominantly rely on scaling laws to improve speech synthesis quality, albeit at the expense of larger model sizes [2]. Conversely, there remains a relatively sparse exploration of lightweight TTS approaches [3, 4]. Going forward, due to limitations in terms of internet access or strict privacy requirements, we believe that the demand for on-device speech synthesis will increase, which would require real-time and on-premise TTS models.

Moreover, as the demand for more personalised and versatile conversational agents grows, the need for adaptable models capable of handling diverse linguistic contexts becomes paramount. This ties in with the heated research avenue of cross-lingual voice cloning [5] in TTS systems, where the goal is to transfer voices across languages. For example, some scenarios would require synthesising Mandarin speech using an English speaker's voice, where the English speaker does not have any Mandarin speech data. Existing solutions tackle this problem by training a large multi-speaker multilingual model with non-polyglot speakers (i.e. each speaker only speaks one language) [6, 7, 8]. Given a large-enough number of examples, such models are able to disentangle speaker and language representations, thus enabling inference for unseen combinations of speaker and language. Other approaches involve the use of a Voice Conversion (VC) model as either a pre-processing or post-processing step. In the former case, cross-lingual VC is used to generate synthetic data of a target speaker speaking in a target language. Then, such synthetic data are used to train a single-speaker monolingual TTS system [9]. In the latter case, cross-lingual VC is used to convert the speaker identity of a speaker in the target language, modeled by a TTS system, into the target speaker identity [10].

Outside of TTS, the realm of Natural Language Processing (NLP) has also experienced notable progress. Within the vast landscape of models, adapters [11, 12] have emerged as a crucial technique for achieving efficient fine-tuning of pre-trained Large Language Models (LLMs) [13]. Serving as compact, learnable parameter modules plugged on top of existing neural network architectures, they enable targeted adaptation without necessitating extensive retraining. As the quest for efficient and robust NLP solutions continues to evolve, different adapter solutions have been proposed for task adaptation [14], language adaptation [15] and more [16].

In this work, we propose to look at cross-lingual voice cloning through the lens of adapters. We believe that adapters, when introduced to TTS systems, present a promising path for addressing one of the key challenges of adaptable systems, namely catastrophic forgetting [17]. Moreover, the concept of adapters has been already introduced to the TTS community, especially for parameter-efficient few-shot speaker adaptation [18, 19, 20]. In this paper, our focus lies in demonstrating the feasibility and efficacy of adapters in the context of lightweight TTS architectures, for both cross-lingual speaker adaptation and language adaptation.

The main contributions of this paper are as follows.

1. We are the first to explore adapters in the context of lightweight, end-to-end (E2E) TTS models in a Generative Adversarial Network (GAN) training setting.

979-8-3503-2411-2/25/$31.00

2. We are the first to propose the use of adapters for TTS language adaptation, where a single-speaker monolingual model is fine-tuned to learn a new language.
3. We propose a new objective method to evaluate accent similarity of a TTS systems against recordings of a native speaker in a target language.

The rest of the paper is organised as follows. In Section 2 we introduce the backbone TTS model and the type of adapters we employ. In Section 3 we present the objective metrics we rely on to evaluate our models. Section 4 contains details about our experiments and results, while Section 5 includes discussion points and concluding remarks.

2. MODEL BACKBONE AND ADAPTERS

The base architecture used in this work is the one presented in [21], which is named Lightweight E2E-TTS (LE2E) and is shown in Fig. 1.

2.1. Model description

The model follows a GAN-based setup with a generator G consisting of an acoustic model and a neural vocoder. The acoustic model, inspired by LightSpeech [3], generates unsupervised acoustic latents from phonemes and positional embeddings. It includes a text encoder, variance adaptor, and acoustic decoder. The neural vocoder, based on Multi-Band MelGAN (MB-MelGAN) [4], converts these latents into waveform signals using up-sampling blocks and residual blocks. The discriminator set D_k includes Multi-Period Discriminators (MPD) [22] and Multi-Resolution Discriminators (MRD) [23].

The model is optimised to minimise the total loss in (1) and the discriminator loss in (2), where the main adversarial training objective, expanded in (2), follows the least squares loss functions for non-vanishing gradient flows [24].

$$\begin{aligned}\mathcal{L} = \mathcal{L}_{dur} + \mathcal{L}_{f0} + \mathcal{L}_G + \lambda_{FM}\mathcal{L}_{FM} + \\ + \lambda_{mel}\mathcal{L}_{mel} + \lambda_{STFT}\mathcal{L}_{STFT}\end{aligned} \tag{1}$$

$$\begin{aligned}\mathcal{L}_G(\hat{x}) &= \min_G \mathbb{E}_{\hat{x}}\left[(D_k(\hat{x}-1)^2\right] \\ \mathcal{L}_D(x,\hat{x}) &= \min_{D_k} \mathbb{E}_x\left[(D_k(x-1)^2\right] + \mathbb{E}_{\hat{x}}\left[(D_k(\hat{x})^2\right]\end{aligned} \tag{2}$$

In terms of variance modelling, $\mathcal{L}_{dur}(d,\hat{d})$ refers to a frame-level Mean-Squared Error (MSE) between predicted $\vec{d}$ and ground-truth durations $\hat{d}$ (extracted from a pre-trained forced aligner), while $\mathcal{L}_{f0}(p,\hat{p})$ is a cross-entropy loss between predicted pitch bins $\mathrm{softmax}(p)$ and ground truth ones $\hat{p}$ (the result of 256-bin quantisation of standardised pitch).

Additional loss functions include a feature matching loss $\mathcal{L}_{FM}$ [24], a multi-resolution STFT loss $\mathcal{L}_{STFT}$ [25] and a mel-spectrogram loss $\mathcal{L}_{mel}$ [22].

2.2. Adapters

As introduced in Section 1, adapters are a computationally-efficient solution to alter the performance of a pre-trained model towards a specific task or domain. Formally, consider a model with parameters θ (pre-trained and frozen) and ϕ (newly introduced). Efficient fine-tuning methods focus on optimising only ϕ based on loss L and dataset D, as reported in (3).

$$\phi_\star \leftarrow \underset{\phi}{\mathrm{argmin}}\, L(D;\{\theta,\phi\}) \tag{3}$$

Efficient fine-tuning techniques, such as bottleneck adapters [11], may insert parameters ϕ at various positions of the pre-trained model. Adapters include a down-projection matrix W_{down}, a non-linearity function f, an up-projection matrix W_{up}, and a residual connection r to transform features h into h', which is formulated as $h' \leftarrow W_{up} \cdot f(W_{down} \cdot h) + r$.

In this work, we make use of two types of adapters: bottleneck [11] and convolutional [26]. For the bottleneck adapter we utilise the vanilla version of it, as shown in Fig. 1(d), setting $f = ReLU$ and the bottleneck dimension to $b_{dim} = 16$. Early experiments revealed that looser bottlenecks $b_{dim} > 16$ were detrimental for speaker/language adaptation.

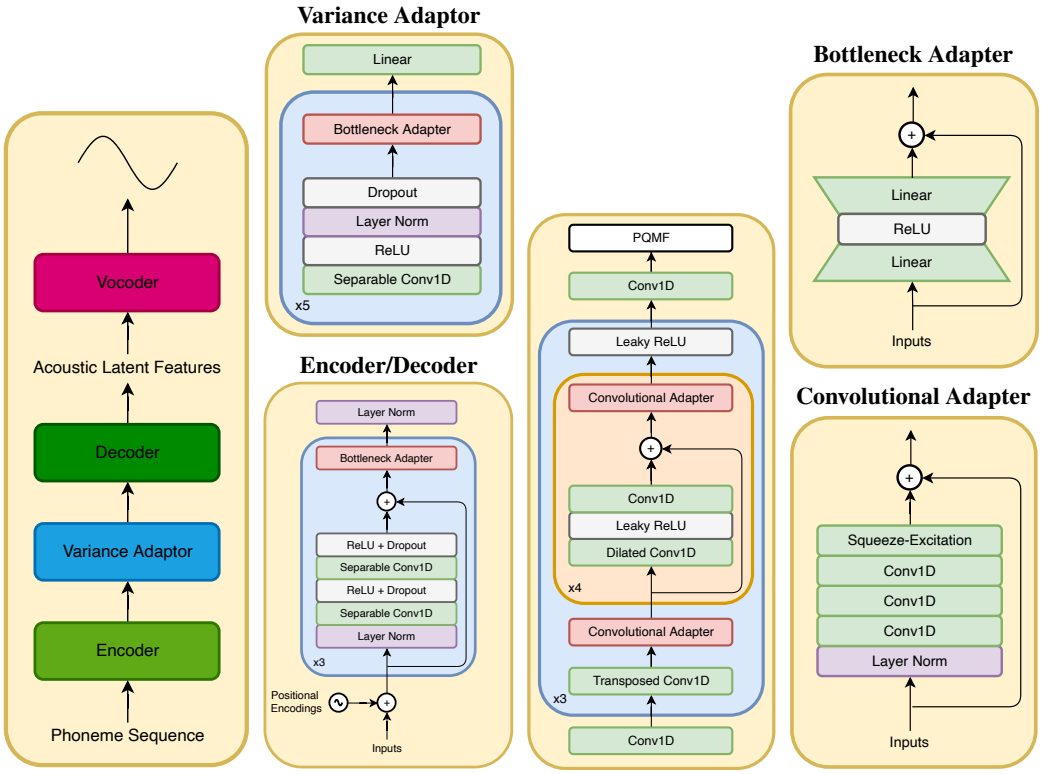
Convolutional adapters were introduced in [26] for extremely efficient task adaptation, where authors show that they outperform bottleneck adapters and achieve comparable performance to LoRA [16] and prefix tuning, with only 0.94% of their trainable parameters, which fits well with our requirements for lightweight TTS. We follow the original implementation, as shown in Fig. 1(d), using three 1D convolutional layers with kernel size $k = [3,5,3]$ (the one with $k = 5$ is a depth-wise convolution [27]) along with layer normalisation and a Squeeze-and-Excitation (SE) module [28]. The output of the SE layer is then summed to the residual connection.

3. EVALUATION METRICS

In the TTS field, model evaluations are tipically done via subjective tests, such as MUSHRA [29]. It is well known that human-based evaluations through crowd-sourcing platforms are a costly, lengthy, and unreliable process where factors like cheaters are guaranteed to bias results [30, 31, 32, 33]. Thus, to assess the quality of generated voices, we discard subjective metrics and only employ objective ones. All evaluations are conducted with 1000 unseen utterances.

3.1. Speaker similarity

We compute the Speaker Embedding Cosine Similarity (SECS) [34] by comparing reference (i.e. voice of the target speaker) and synthetic audios using a d-vector speaker verification model [35] to gauge speaker similarity. SECS involves the computation of cosine similarity (in the $[-1,1]$ range,



(a) Entire model (b) Acoustic model (c) Vocoder (d) Adapters

Fig. 1: Model architecture at inference time. At training time, discriminators are also present.

with 1 indicating "same speaker") between the embeddings of two audios that are extracted from a pre-trained speaker encoder.

3.2. Signal quality, intelligibility, naturalness

We evaluate signal quality, intelligibility, and naturalness using TorchAudio-squim [36] metrics. In particular, we use a combination of Perceptual Evaluation of Speech Quality (PESQ) and Scale-Invariant Signal-to-Distortion Ratio (SI-SDR) for signal quality; Short-Time Objective Intelligibility (STOI) for intelligibility; and NORESQA-MOS (which we will refer to as MOS) for naturalness.

3.3. Accent nativeness

For assessing the nativeness of speakers in a target language, we propose a new objective metric for more reliable and reproducible accent comparisons. We base this metric on an Automatic Speech Recognition (ASR) model. In the field of Computer-Assisted Pronunciation Training (CAPT), various ASR-based systems have been proposed for detecting mispronunciation in speech uttered by second-language (L2) learners [37, 38]. We refer to this task as Mispronunciation Detection and Diagnosis (MDD) [39]. There are different ways to tackle this problem, but the most effective ones directly transcribe phoneme-level symbols [38] from speech signals, which are then used for measuring mispronunciations.

The MDD task targeted at L2 learners aligns perfectly with our goal of accent evaluation. Therefore, in this paper we use objective metrics produced by such MDD models for comparing the nativeness of accent in generated speech from different systems. In particular, the MDD model we use is very similar to the one introduced in [39], which is based off of wav2vec2 [40]. Our approach entails stacking a classification head on top of a large pre-trained multilingual wav2vec2 model[1] and fine-tuning it with a Connectionist Temporal Classification (CTC) loss for the phoneme recognition task. As a proxy for accent similarity, we calculate the Phoneme Substitution Rate (PSR), i.e. the percentage of phonemes in an utterance which are substituted with a phoneme different from the ground truth. Numerous prior studies [41, 42] have shown that phoneme substitution is a key pattern in accented speech. Therefore, we consider it as a reasonable objective metric for measuring non-native accent, as a higher phoneme substitution rate correlates with higher level of non-native accent.

We validate this approach by comparing our historical human accent MUSHRA test results with the results produced by our MDD model, which confirmed that we are able to pro-

[1] https://huggingface.co/facebook/wav2vec2-large-xlsr-53

duce consistent rankings amongst different systems as human testers did. To do so, we compare pairs of systems from completed MUSHRA tests and compute the PSR of both systems in the pair. If the PSR for the system that has the highest MUSHRA score in the pair is lower than that of the other system in the pair, we mark this as a successful classification from the MDD model. We compare more than 50 pairs of systems in different languages (`de-DE`, `en-AU`, `en-GB`, `es-US` and `fr-CA`) and obtain an average accuracy of more than 90%.

4. EXPERIMENTS AND RESULTS

In this section, we formally introduce the tasks of speaker and language adaptation and present our experimental results on both when comparing full fine-tuning (i.e. without adapters, fine-tuning the entire model) and adapter-based fine-tuning scenarios (i.e. with adapters, only fine-tuning adapters).

4.1. Adapters settings

We rely on bottleneck adapters for the LightSpeech-based encoder, decoder, duration and pitch predictors. We place one adapter block after each convolutional layer, as shown in Fig. 1(d), and we set the bottleneck dimension to $b_{dim} = 16$ for all adapter blocks, which gives us $150K$ additional trainable parameters in the acoustic model. On the vocoder side, we opt for convolutional adapters and we position one block after each upsampling layer and residual block, for a total of 15 blocks: 5 for each upsampling block, 1 after the transposed convolution and one after each of the 4 residual blocks. This gives us another $50K$ additional trainable parameters, for a total of $200K$ parameters, which corresponds to 10% of the original model's capacity ($2M$ parameters). We experimented with using bottleneck adapters in the entire model, but opted for convolutional ones for the vocoder to keep the number of parameters under control, given the number of adapter modules needed in the vocoder to achieve high-quality adaptation.

4.2. Training settings

All backbone models (described in Sections 4.3 and 4.4) underwent training for $1M$ steps, with an effective batch size of 128 samples. The training utilized the AdamW optimizer [43] with $\beta_1 = 0.8$ and $\beta_2 = 0.99$, along with an initial learning rate set at 2×10^{-4} and an exponential decay scheduler with a factor of $\gamma = 0.99$. Additionally, a weight decay penalty factor of 1×10^{-2} was applied. For a more in-depth understanding, readers are directed to [21].

During fine-tuning, models are optimised by freezing all modules with the exception of adapter blocks, the phoneme embedding matrix and the speaker/language encoding look-up-tables. Losses are the same as those used for training the backbone models and all fine-tuning experiments are run for 200 epochs with the same optimiser and LR scheduler used for the backbone models, with the only difference in the initial LR for the full fine-tuning version and the adapter-based fine-tuning, equal to 1×10^{-5} and 1×10^{-4}, respectively. This is because adapters contain newly introduced parameters and we find it beneficial to favour a higher degree of exploration.

4.3. Language adaptation

Given a TTS model pre-trained on speaker S_1 (where S_1 speaks in language L_1), we define language adaptation as adapting it to speak in a new, unseen language L_2 while maintaining the speaker identity of S_1. In this work, we pre-train the backbone model described in Section 2.1 on roughly 10 hours of recordings from a native British English (`en-GB`) speaker (S_1). Our goal is to adapt this pre-trained model to Castillan Spanish (`es-ES`) and we do so by fine-tuning the model on 100 hours of `es-ES` data that spans across $n = 25$ different speakers $S_2, \dots S_{n+1}$. The setup of adapters and the training settings are reported in Sections 4.1 and 4.2.

In Table 1 we show results comparing full fine-tuning ($\star_1$) and adapter-based fine-tuning ($\star_2$). What we observe is that adapter-based fine-tuning results in overall better signal quality, intelligibility and naturalness, as indicated by PESQ, SI-SDR, STOI and MOS metrics, higher speaker similarity to the target speaker S_1 and a slightly lower accent nativeness w.r.t. the full fine-tuning scenario. Informal listening shows that the lower accent nativeness in the adapter-based scenario is due to full fine-tuning overfitting on one of the `es-ES` speakers in the dataset, which explains the lower SECS score.

To understand the effect of adapter placement on its efficacy, we reduce the number of convolutional adapters in the vocoder from 15 to 3, removing all adapters after residual blocks. In Table 1 ($\star_3$) we can see a slight degradation for both speaker similarity and accent nativeness metrics. Similarly, in Table 1 ($\star_4$) we show that relying on only 1 `es-ES` speaker (instead of n) for fine-tuning (with adapters) the TTS model to language L_2 results in severe degradation in accent nativeness, with an expected improvement in speaker similarity (due to English accent leakage from speaker S_1).

4.4. Speaker adaptation

Given a TTS model pre-trained on a set of n speakers $S_2, \dots S_{n+1}$ (where $S_2, \dots S_{n+1}$ all speak in language L_2), we define speaker adaptation as adapting it to sound like a new, unseen speaker S_1 (which speaks in language L_1). We use the same dataset from Section 4.3, but in the opposite way: the `es-ES` data is used for training the backbone model, while the `en-GB` data is used for adapting the speaker identity.

In Table 2 we show results comparing full fine-tuning and adapter-based fine-tunings for the speaker adaptation scenario. In particular, we train two model variants with adapters added to different modules, i.e. one with adapters added only in the vocoder ($\dagger_2$) and another one with adapters

Table 1: Results for the language adaptation task.

	SECS (↑)	**PESQ** (↑)	**SI-SDR** (↑)	**STOI** (↑)	**MOS** (↑)	**PSR** (↓)
$\star_1$ Full fine-tuning	0.43 ± 0.04	3.15 ± 0.26	20.81 ± 2.65	0.99 ± 0.00	3.99 ± 0.17	$\mathbf{0.56 \pm 1.33}$
$\star_2$ Adapters fine-tuning	0.51 ± 0.04	3.29 ± 0.25	19.72 ± 2.20	0.99 ± 0.00	$\mathbf{4.36 \pm 0.14}$	1.42 ± 1.95
$\star_3$ Less adapters in vocoder	0.49 ± 0.05	$\mathbf{3.57 \pm 0.26}$	$\mathbf{21.75 \pm 2.97}$	0.99 ± 0.00	3.61 ± 0.42	1.47 ± 2.08
$\star_4$ Single `es-ES` speaker	$\mathbf{0.63 \pm 0.04}$	3.38 ± 0.26	21.34 ± 2.61	0.99 ± 0.00	4.23 ± 0.23	5.81 ± 4.53

Table 2: Results for the speaker adaptation task.

	SECS (↑)	**PESQ** (↑)	**SI-SDR** (↑)	**STOI** (↑)	**MOS** (↑)	**PSR** (↓)
$\dagger_1$ Full fine-tuning	0.70 ± 0.05	2.51 ± 0.36	15.96 ± 3.32	0.98 ± 0.00	3.87 ± 0.37	8.44 ± 5.41
$\dagger_2$ Adapters fine-tuning	0.62 ± 0.05	$\mathbf{3.28 \pm 0.26}$	20.15 ± 2.12	0.98 ± 0.00	4.19 ± 0.25	$\mathbf{0.75 \pm 1.56}$
$\dagger_3$ Adapters in entire model	$\mathbf{0.78 \pm 0.04}$	2.98 ± 0.35	$\mathbf{20.40 \pm 1.82}$	$\mathbf{0.99 \pm 0.00}$	$\mathbf{4.23 \pm 0.22}$	6.27 ± 3.80
$\dagger_4$ Single `es-ES` speaker	0.58 ± 0.04	1.16 ± 0.03	0.30 ± 1.50	0.79 ± 0.03	2.53 ± 0.12	43.68 ± 6.58

additionally added in the acoustic model ($\dagger_3$). Similar to before, we have as one comparison baseline the full-fine-tuning scenario without adapters ($\dagger_1$). Moreover, we pre-train the TTS model on only one es-ES speaker (instead of n) and then fine-tune it (with adapters) to speaker S_1 for exploring the effect of reducing the amount of training data in L_2 ($\dagger_4$).

Overall, we observe that adapter-based fine-tuning achieves the best results across all axes. Looking closer, we can see that removing adapters from the acoustic model leads to significantly better accent nativeness, reflected in almost ten-fold reduction in PSR. This indicates that the model encodes the majority of language-agnostic speaker information in the vocoder modules. All metrics considered, we conclude that vocoder-only adapter fine-tuning achieves the best outcome. On the other hand, it is also clear that reducing the amount of training data in L_2 during pre-training of the TTS model does not favour speaker adaptation. This suggests that the model must learn meaningful speaker representations in the pre-training phase for achieving high-quality speaker adaptation.

5. CONCLUSIONS

In this paper, we explored the application of adapters in the context of E2E lightweight TTS, with a particular focus on cross-lingual speaker adaptation and language adaptation. Our investigation shows that adapters are a viable solution to adapt pre-trained TTS models to unseen languages or speakers, with only 10% additional model parameters. Comparing speaker and language adaptations, we find the former to be an easier task for target models, even though the latter results in the best combination of objective metrics when the goal is to transfer voices across languages. Finally, our experiments reveal that both strategic positioning of adapters in the vocoder and speaker variety in the target language play a crucial role for accent nativeness and speaker similarity.

6. REFERENCES

[1] J. Betker, "Better speech synthesis through scaling," 2023.

[2] C. Wang, S. Chen, Y. Wu, and Zothers, "Neural codec language models are zero-shot text to speech synthesizers," 2023.

[3] R. Luo, X. Tan, R. Wang, et al., "Lightspeech: Lightweight and fast text to speech with neural architecture search," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 5699–5703.

[4] G. Yang, S. Yang, K. Liu, et al., "Multi-band melgan: Faster waveform generation for high-quality text-to-speech," in *2021 IEEE Spoken Language Technology Workshop (SLT)*, 2021, pp. 492–498.

[5] Z. Zhang, L. Zhou, C. Wang, et al., "Speak foreign languages with your own voice: Cross-lingual neural codec language modeling," 2023.

[6] Z. Ziyao, F. Alessio, S. Ariadna, et al., "Mix and Match: An Empirical Study on Training Corpus Composition for Polyglot Text-To-Speech (TTS)," in *Proc. Interspeech 2022*, 2022, pp. 2353–2357.

[7] Y. Zhang, R. J. Weiss, H. Zen, et al., "Learning to Speak Fluently in a Foreign Language: Multilingual Speech Synthesis and Cross-Language Voice Cloning," in *Proc. Interspeech 2019*, 2019, pp. 2080–2084.

[8] E. Nachmani and L. Wolf, "Unsupervised polyglot text-to-speech," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

[9] L. Finkelstein, H. Zen, N. Casagrande, et al., "Training Text-To-Speech Systems From Synthetic Data: A Practical Approach For Accent Transfer Tasks," in *Proc. Interspeech 2022*, pp. 4571–4575.

[10] N. Ellinas, G. Vamvoukakis, K. Markopoulos, et al., "Cross-lingual text-to-speech with flow-based voice conversion for improved pronunciation," 2022.

[11] N. Houlsby, A. Giurgiu, S. Jastrzebski, et al., "Parameter-efficient transfer learning for NLP," in *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp. 2790–2799.

[12] J. He, C. Zhou, X. Ma, et al., "Towards a unified view of parameter-efficient transfer learning," in *International Con-*

ference on Learning Representations, 2022.

[13] T. Brown, B. Mann, N. Ryder, et al., "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, 2020.

[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)*, Aug. 2021.

[15] J. Pfeiffer, I. Vulić, I. Gurevych, et al., "MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

[16] E. J. Hu, Y. Shen, P. Wallis, et al., "LoRA: Low-rank adaptation of large language models," in *International Conference on Learning Representations*, 2022.

[17] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, et al., "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, vol. 114, 12 2016.

[18] N. Morioka, H. Zen, N. Chen, et al., "Residual adapters for few-shot text-to-speech speaker adaptation," 2022.

[19] 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.

[20] A. Mehrish, A. Ramesh Kashyap, L. Yingting, et al., "ADAPTERMIX: Exploring the Efficacy of Mixture of Adapters for Low-Resource TTS Adaptation," in *Proc. INTERSPEECH 2023*, 2023, pp. 4284–4288.

[21] B. T. Vecino, A. Gabrys, D. Matwicki, et al., "Lightweight End-to-end Text-to-speech Synthesis for low resource on-device applications," in *Proc. 12th ISCA Speech Synthesis Workshop (SSW2023)*, 2023, pp. 225–229.

[22] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17022–17033, 2020.

[23] W. Jang, D. Lim, J. Yoon, et al., "UnivNet: A Neural Vocoder with Multi-Resolution Spectrogram Discriminators for High-Fidelity Waveform Generation," in *Proc. Interspeech 2021*.

[24] K. Kumar, R. Kumar, T. De Boissiere, et al., "Melgan: Generative adversarial networks for conditional waveform synthesis," *Advances in neural information processing systems*, vol. 32, 2019.

[25] R. Yamamoto, E. Song, and J.-M. Kim, "Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

[26] Y. Li, A. Mehrish, R. Bhardwaj, et al., "Evaluating parameter-efficient transfer learning approaches on sure benchmark for speech understanding," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[27] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[28] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7132–7141.

[29] ITUR Recommendation, "Bs. 1534-1. method for the subjective assessment of intermediate sound quality (mushra)," *International Telecommunications Union, Geneva*, 2001.

[30] R. Zequeira Jiménez, A. Llagostera, B. Naderi, et al., "Intra- and inter-rater agreement in a subjective speech quality assessment task in crowdsourcing," in *Companion Proceedings of The 2019 World Wide Web Conference*, 2019, pp. 1138–1143.

[31] S. Buchholz and J. Latorre, "Crowdsourcing preference tests, and how to detect cheating," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[32] A. Burmania, S. Parthasarathy, and C. Busso, "Increasing the reliability of crowdsourcing evaluations using online quality assessment," *IEEE Transactions on Affective Computing*, vol. 7, no. 4, pp. 374–388, 2015.

[33] R. Z. Jiménez, L. F. Gallardo, and S. Möller, "Influence of number of stimuli for subjective speech quality assessment in crowdsourcing," in *2018 Tenth international conference on quality of multimedia experience (QoMEX)*. IEEE, 2018, pp. 1–6.

[34] E. Casanova, C. Shulby, E. Gölge, et al., "SC-GlowTTS: An Efficient Zero-Shot Multi-Speaker Text-To-Speech Model," in *Proc. Interspeech 2021*, 2021, pp. 3645–3649.

[35] L. Wan, Q. Wang, A. Papir, et al., "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4879–4883.

[36] A. Kumar, K. Tan, Z. Ni, et al., "Torchaudio-squim: Reference-less speech quality and intelligibility measures in torchaudio," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[37] B.-C. Yan, M.-C. Wu, H.-T. Hung, et al., "An end-to-end mispronunciation detection system for l2 english speech leveraging novel anti-phone modeling," *arXiv preprint arXiv:2005.11950*, 2020.

[38] T.-H. Lo, S.-Y. Weng, H.-J. Chang, et al., "An effective end-to-end modeling approach for mispronunciation detection," *arXiv preprint arXiv:2005.08440*.

[39] L. Peng, K. Fu, B. Lin, et al., "A study on fine-tuning wav2vec2. 0 model for the task of mispronunciation detection and diagnosis.," in *Interspeech*, 2021, pp. 4448–4452.

[40] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, et al., Eds. 2020, vol. 33, pp. 12449–12460, Curran Associates, Inc.

[41] S. Mao, X. Li, K. Li, et al., "Unsupervised discovery of an extended phoneme set in l2 english speech for mispronunciation detection and diagnosis," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 6244–6248.

[42] X. Li, S. Mao, X. Wu, et al., "Unsupervised Discovery of Non-native Phonetic Patterns in L2 English Speech for Mispronunciation Detection and Diagnosis," in *Proc. Interspeech 2018*, 2018, pp. 2554–2558.

[43] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2019.