Interspeech 2024
1-5 September 2024, Kos, Greece

Personality-memory Gated Adaptation: An Efficient Speaker Adaptation for Personalized End-to-end Automatic Speech Recognition

Yue Gu[1], *Zhihao Du, Shiliang Zhang, Jiqing Han*[1*], *Yongjun He*[1]

[1]School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China
yuegu@stu.hit.edu.cn, jqhan@hit.edu.cn

Abstract

In real-world applications, a preferred personalized automatic speech recognition (ASR) model should exhibit robust recognition capabilities that include both personalization and generalization. Speaker adaptation is a commonly used approach towards personalized ASR. However, most speaker adaptation methods only focus on improving the performance of the target speaker but neglect or even sacrifice generalization. In this paper, we propose the Personality-memory Gated Adaptation (PGA) approach to adapt models on target speakers while preserving generalized recognition performance. Specifically, we incorporate parallel adapters into the encoder to capture the target speaker's vocal characteristics. The hidden outputs of the adapters and original encoder layers are fused via a scalar gate, which is derived from the similarity between input samples and personality-memory units, i.e., personality embeddings from the target speaker. In this manner, the vocal characteristics of the target speaker are memorized by adapters while the ASR backbone preserves generalization. Experimental results show that, compared to the unadapted ASR backbone, the proposed PGA achieves 21.62% relative character error rate (CER) reduction on the target domain but a negligible CER increase on the source domain with limited adaptation data and training steps[1].
Index Terms: speaker adaptation, memory gated, personalized speech recognition

1. Introduction

From the perspective of practical applications, a preferred personalized ASR model is expected to exhibit robust recognition capabilities, specifically demonstrating powerful personalized recognition and generalized recognition capabilities.

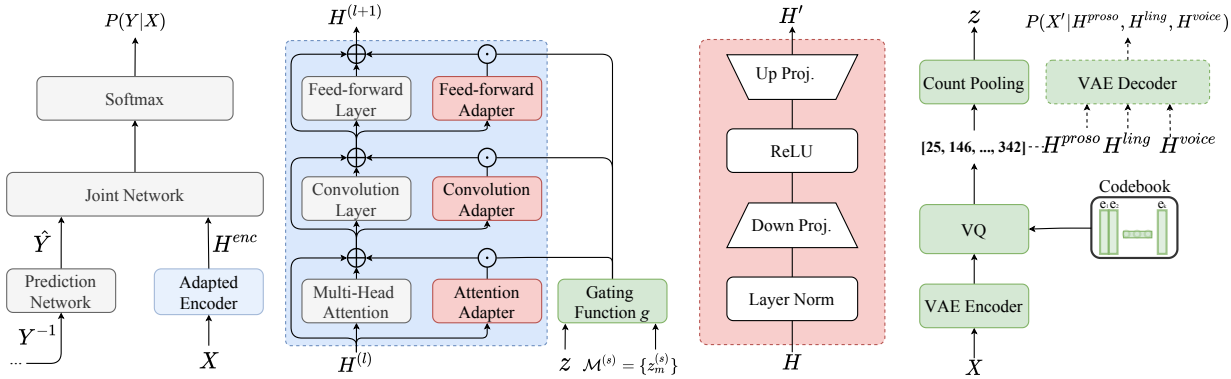
Speaker adaptation is a hot topic related to personalized ASR which aims at alleviating the acoustic mismatch between the training domain and target speakers. Speaker adaptation methods can be broadly categorized into embedding-based, data augmentation, and model-based approaches [1]. The paradigm of the first category is incorporating the speaker embedding [2, 3] or speaker code [4, 5] to the input feature [6, 7], hidden layer input [8, 9], or attention module [10, 11, 12] for Speaker Adaptive Training (SAT). SAT-based methods introduce speaker embedding extractor to ASR, achieving moderate improvement on the target domain [13]. Both of the latter two categories can benefit from adaptation data collected from the target speaker. In data augmentation approaches, a text-to-speech synthesis model is first fine-tuned with adaptation data and then used to generate numerous synthesized utterances for speaker adaptation [14, 15]. In model-based approaches, a commonly used method is fine-tuning ASR models with target-speaker adaptation data, which is also named speaker signature in [14]. Other model-based approaches [16, 17, 18, 19], such as LHUC, can be regarded as a case of parameter isolation in transfer learning, which are designed to leverage the generic features learned from the source domain while being adapted to the target domain, and the parameters of the source and target modules are kept separately. Recently, Parameter-Efficient Fine-Tuning (PEFT) has been one of the mainstream approaches within the category of parameter-isolation techniques in transfer learning. PEFT demonstrates better adaptation ability than LHUC [20] and is introduced to model-based speaker adaptation. In PEFT, adapter modules with small-scale parameters are inserted between layers of a pre-trained model, allowing task-specific customization without altering the original model parameters [21, 22, 23]. The effectiveness of adapters has been proven for atypical and accent adaptation [13], dialect identification [24], and domain adaptation [25].

Model-based methods have the potential to achieve better performance with specific adaptation parameters and data, attracting more research attention. However, all above mentioned model-based methods cannot maintain personalized and generalized recognition capabilities in a single ASR model, simultaneously. To improve the adaptation performance as much as possible while preserving the original generalization capability, Personality-Aware Training (PAT) [26] and Adapter/Cautious-Fine-Tuning (A/CFT) [27] are introduced. While PAT employs the personality similarity to reweight training samples for fine-tuning, A/CFT first trains the adapter and then finetunes the model with a cautious learning rate on the adaptation data. Although PAT and A/CFT achieve promising results, they require either numerous training steps or a large amount of adaptation data.

In this paper, we propose an efficient speaker adaptation method named Personality-memory Gating Adaptation (PGA). Specifically, we incorporate parallel adapters into the ASR encoder to capture the target speaker's vocal characteristics. The hidden outputs of the adapters and original encoder layers are fused via a scalar gate, which is derived from the similarity between input samples and personality-memory units, i.e., personality embeddings from the target speaker. In this manner, the vocal characteristics of the target speaker are memorized by adapters while the ASR backbone preserves generalization. Therefore, only a small number of adaptation data and training steps are required to find the balance between them with the gating function, resulting in efficient adaptation processing. Experimental results show that the proposed PGA significantly improves the performance of the target speaker and preserves generalized recognition capability with limited adaptation training steps and 10-minute adaptation data.

[1]github:shibeiing/Personality-memory-Gated-Adaptation-PGA

10.21437/Interspeech.2024-1621



(a) The adapted RNN-T model (b) The architecture of l-th adapted encoder block (c) The design of adapter (d) The personality embedding extractor

Figure 1: *Schematic diagram of the proposed personality-memory gated adapter (PGA) in (a) the adapted RNN-T model. (b) provides an enlarged view of each encoder block equipped with PGA. (c) demonstrates the shared design of adapters for attention, convolution, and feed-forward modules. (d) illustrates the personality extractor at both inference (solid) and training (dashed) stages.*

2. Personality-memory Gated Adaptation

2.1. Overview

In PGA, on the one hand, a well-trained ASR model is employed as the backbone to model generalized information among a rich variety of speakers, accents, channels, and acoustic environments. On the other hand, as shown in Figure 1(b), we incorporate adapters into ASR encoder layers to memorize the personalized information of the target speaker. The parallel fusion facilitates to balance the personalized information and generalized information and exhibits better adaptation performance than series fusion [28]. In parallel fusion, we design a gating function to balance the generalized projections from the ASR backbone and personalized projections from adapters. The result of the gating function is a scalar indicating the acoustic similarity between input utterance and the personality-memory units, which are described in sections 2.1 and 2.2. Considering that linguistic information is shared among different speakers while the acoustic characteristic varies, we apply our PGA on the encoder blocks and keep the architecture of the prediction and joint networks unchanged. Thus, the proposed PGA can also be applied to other ASR models easily, such as Attention Encoder-Decoder (AED) and Connectionist Temporal Classification (CTC). Figure 1 provides an overview of our PGA.

2.2. Personality Extraction of the Target Speaker

In PAT [26], the phone-aligned prosody embedding extractor exhibits comparable performance but has a smaller footprint than voice-print vectors, therefore, in this paper, we employ the prosody embeddings to represent the personality of the target speaker. As shown in Figure 1(d), the prosody extractor is based on Vector Quantized Variational AutoEncoder (VQ-VAE) [29] which involves Vector Quantization (VQ) into VAE. The prosody extractor mainly comprises three parts at the training stage: an encoder to map input features X into latent representations, a VQ module to obtain discrete prosody tokens with an exponentially updated codebook, and a decoder to reconstruct the spectrogram X' with the prosody tokens H^{proso}, the linguistic embeddings H^{ling} and the voice-print embedding H^{voice}. H^{ling} is the phonetic posteriorgram predicted by a phoneme recognition model [30]. Since H^{voice} and H^{ling} represent the timbre and content in speech, respectively, H^{proso} contains much more timbre-and-content-independent prosody information [31]. More training details and settings can be found in [26]. At the inference stage, the prosody tokens H_m^{proso} of utterance m are passed through a count pooling to extract the personality embedding $z_m^{(s)}$ for the target speaker s:

$$z_m^{(s)} = \frac{1}{T}\sum_{t=1}^{T}\mathbb{I}(H_{m,t}^{proso}) \tag{1}$$

where $\mathbb{I}$ denotes a one-hot function with the same size as the VQ codebook. For a target speaker with M utterances, the personality memory $\mathcal{M}^{(s)}$ is constructed by M embeddings: $\mathcal{M}^{(s)} = \{z_1^{(s)}, z_2^{(s)}, \ldots, z_M^{(s)}\}$.

2.3. Personality-memory Gating Function

We perform a layer-wise adaption for each encoder block with a personality-memory-derived gating function. As shown in Figure 1(b), in each conformer-based encoder block, there are three adapters for multi-head attention (*attn*), convolutional layer (*conv*) and feed-forward projection (*ff*). To make the adapter focus on the vocal characteristic of the target speaker while the original encoder modules preserve generalization for others, we propose a personality-memory derived gating function f to merge the outputs of adapters A and original modules H:

$$\begin{aligned}
\bar{H}_{attn}^{(l)} &= H_{attn}^{(l)} + f(z, \mathcal{M}^{(s)}) \cdot A_{attn}^{(l)} + H^{(l)} \\
\bar{H}_{conv}^{(l)} &= H_{conv}^{(l)} + f(z, \mathcal{M}^{(s)}) \cdot A_{conv}^{(l)} + \bar{H}_{attn}^{(l)} \\
\bar{H}^{(l+1)} &= H_{ff}^{(l)} + f(z, \mathcal{M}^{(s)}) \cdot A_{ff}^{(l)} + \bar{H}_{conv}^{(l)}
\end{aligned} \tag{2}$$

where z represents the personality embedding of the input utterance and l is the encoder block index. The gating value is derived from the cosine similarity between the personality embedding z and the memory of the target speaker:

$$f(z, \mathcal{M}^{(s)}) = \sigma\left(\tau\left(2\frac{\alpha}{\theta} - 1\right)\right) \tag{3}$$

$$\alpha = \frac{1}{M}\sum_{z_m^{(s)} \in \mathcal{M}^{(s)}} \frac{< z, z_m^{(s)} >}{||z|| \cdot ||z_m^{(s)}||} \tag{4}$$

where σ denotes the sigmoid function. θ is a hyper-parameter to rescale the cosine similarity, ranging from 0 to 1. τ is a temperature term to increase or decrease the sharpness of the sigmoid function, setting to 5 in our experiments. $< \cdot, \cdot >$ represents the dot product of two vectors, and $||\cdot||$ denotes the L2 norm of a vector. To enrich the training sample variation, the embeddings in personality memory $\mathcal{M}^{(s)}$ are randomly selected from the target speaker and vary in different mini-batches.

2.4. The shared design of Adapters

Adapters of multi-head attention, convolutional layer, and feed-forward projection share the same design. In each adapter, input hidden states H undergo layer normalization, down-dimensional projection, ReLU non-linearity, and up-dimensional projection as follows:

$$\begin{aligned} \tilde{H} &= \text{LayerNorm}(H) \\ H' &= W_{up}\text{ReLU}(W_{down}\tilde{H} + b_{down}) + b_{up} \end{aligned} \tag{5}$$

where $W_{down} \in \mathbb{R}^{r\times d}$ and $W_{up} \in \mathbb{R}^{d\times r}$ are uesed for dimension reduction. d is the same as the output dimension of encoder outputs and r represents the bottleneck dimension of adapters.

2.5. Training

The fine-tuning process with PGA is divided into two stages. In the first stage, we freeze both the ASR backbone and the gating function and train the parallel adapters on the adaptation data of the target speaker. In this manner, the personalized information of the target speaker is memorized in the adapters. In the second stage, we freeze both the learned adapter and the gating function and fine-tune the ASR backbone with the gating function to balance the generalized projections from the backbone and personalized projections from adapters. The backbone parameter is tuned using the adaptation data and all or part of the training data. Note that, the training and adaptation data is only replayed **once** to the model at the second stage, reducing adaptation training steps.

3. Experimental Settings

3.1. Data and Metrics

We evaluate and analyze the proposed PGA on three opening corpus: Kespeech-phrase1 [32], MagicData-Sichuan[2] and MagicData-Zhengzhou[3]. KeSpeech-phrase1 corpus contains 895 hours of transcribed audio in Mandarin (standard Chinese) or Mandarin subdialect from 27,044 speakers in 34 cities across China. In Kespeech-phrase1, Mandarin subdialect refers to 8 distinguishable subdialects far from standard Chinese, and the amount of all subdialect utterances is far less than in Mandarin. For example, the number of utterances in Mandarin is twelve times that of Southwestern Mandarin and fourteen times that of Zhongyuan Mandarin. This imbalance of data is derived from the real world and is well suited to verify the effectiveness of speaker adaptation methods. We refer to the official data division of Kespeech-phrase1 to train the baseline model, where the test set with a duration of 31 hours is considered as the source domain test set. MagicData-Sichuan and MagicData-Zhengzhou are viewed as target domain data, which includes 20 speakers and 30-minute labeled utterances for each speaker. Consistent with [26], we use 6 speakers' data to evaluate the proposed PGA, 10 minutes for target speaker adaptation, and 20 minutes for target speaker testing. Notably, Magicdata-Sichuan belongs to the Southwestern Mandarin and MagicData-Zhengzhou belongs to Zhongyuan Mandarin.

We evaluate the proposed PGA with two aspects: performance and cost. It is realistic to adapt personalized ASR models on the cloud and push them back to personal devices [1, 33]. Thus, not only the character error rate (CER) on the source and target domains but also the number of parameters during inference and the adaptation training steps are considered.

[2]https://magichub.com/datasets/sichuan-dialect-scripted-speech-corpus-daily-use-sentence/

[3]https://magichub.com/datasets/zhengzhou-dialect-scripted-speech-corpus-daily-use-sentence/

Table 1: *Comparison of other speaker-adapted models on the source and target domain test sets: performance (CER, %) and cost (parameters in inference and adaptive training steps).*

Model	Performance		Cost	
	Source	Target	# Params	# Steps
Baseline	12.69	19.06	9.4 M	–
Baseline (large)	–	15.81	112.8 M	–
Spksign [14, 26]	13.14	17.82	9.4 M	20
Spksign-encoder [14]	13.14	18.15	9.4 M	20
SAT-feat [7, 26]	12.79	18.35	53.9 M	–
SAT-feat+Spksign [26]	14.79	18.09	53.9 M	20
SAT-block [8]	12.70	18.00	53.9 M	–
SAT-block+Spksign	13.36	17.28	53.9 M	20
A/CFT [27]	20.36	14.60	10.4 M	22
PAT (speaker) [26]	12.82	16.80	9.4 M	242.1 K
PAT (prosody) [26]	12.63	16.87	9.4 M	242.1 K
PGA	12.96	14.94	11.3 M	4.77 K

3.2. Experimental Setup

Conformer transducer is one of the most popular ASRs, we employ it as the baseline and the ASR backbone of the proposed PGA. As shown in Table 1, there are two different sizes of baseline in our experiments, the large one has ten times of parameters than the small one. This setting takes into account the fact that personalized ASRs are often deployed on resource-constrained personal devices, with the goal that the small adapted model can match or even exceed the performance of the large ASR for the specific person. Notably, the configurations of baseline and baseline (large) are the same as Conformer (S) and Conformer (L) in [34]. We use the ESPnet toolkit [35] to implement our PGA and follow its default recipe settings (same as egs/aishell recipe). The personality embedding extractor model and the speaker embedding extractor are open-released in [26], which can be used in our experiments directly.

With the pre-trained ASR backbone and personality embedding extractor initialization, we freeze them and train the adapter modules with adam optimizer with a learning rate of 0.0015 for 20 steps. The output dimension of encoder d is 144, and the bottleneck dimension r is set to 108. Then, we use gated fine-tuning to optimize the PGA model on all training data and adaptation data for 1 epoch (4.75 K steps) with a small learning rate of 5e-7. The amounts of parameters in the adapter module and personality embedding extractor are about 1 M.

4. Results

In this section, we compare some adaptation methods with the proposed PGA in both performance and cost. Speaker signature (Spksign) [14] is the most common adaptation method for end-to-end ASR, in which a limited adaptation data is used to fine-tune part or all parameters of the pre-trained speaker-independent ASR model. In our experiments, Spksign means fine-tuning the whole model parameters, and Spksign-encoder means only fine-tuning the encoder parameters, both of them using a small learning rate of 1e-5 which is consistent with [14]. Speaker Adaptive Training (SAT) incorporates the pre-trained speaker embedding extractor to ASR for training speaker-dependent ASR. SAT-feat refers to concatenating the speaker embedding to the input feature [7], and SAT-block refers to concatenating the speaker embedding to the conformer block input [8]. SAT-feat+Spksign and SAT-block+Spksign refer to

Table 2: *Ablation study on PGA: performance (CER, %).*

	adapter	gating function	fine-tune	Source	Target
Baseline	-	-	-	12.69	19.06
PGA	✓	✓	✓	12.96	14.94
EXP1	✓		✓	13.99	15.80
EXP2	✓			20.27	14.60
EXP3			✓	13.14	18.15

Table 3: *Investigation on the effect of the amount of data for gated fine-tuning: performance (CER, %) and cost (parameters in inference and adaptive training steps). 'proportion' means the portion of training data used for gated fine-tuning.*

proportion	Performance		Cost	
	Source	Target	# Params	# Steps
100%	12.96	14.94	11.3 M	4770
10%	13.18	15.02	11.3 M	508
1%	13.22	15.01	11.3 M	72

fine-tuning the SAT-based speaker-dependent ASR model with adaptation data for the target speaker. Then, Adapter/Cautious-Fine-Tuning (A/CFT) [27] first trains the adapters on a new task while freezing the ASR backbone then adapts the entire model with a minimal learning rate using only adaptation data. The Personality-Aware Training (PAT) [26] uses speaker and prosody embedding similarity between train and adaptation data to reweight the ASR loss terms.

4.1. Comparison with other Speaker Adaptation Methods

As shown in Table 1, all adaptation methods improve the performance on the target domain. Compared with the baseline, PGA achieves relative CER reductions of 21.62% on the target domain and a negligible CER increase of 0.27% on the source domain with limited external parameters and training steps. Surprisingly, PGA achieves better performance on the target domain than baseline (large), the latter has ten times more parameters than the former. This indicates that it is reasonable to employ small-footprint personal ASR models instead of the large version on personal devices. As a continual learning method, the amounts of new task utterances for A/CFT may be hundreds of thousands, but this is nearly unachievable for speaker adaptation. Although A/CFT achieves the best performance on the target domain, the CER of the source domain deteriorates heavily. PAT (prosody) is the most comparable adaptation method which achieves a trade-off between source domain and target domain and does not involve external parameters. However, it needs hundreds of thousands of training steps to adapt the ASR model for the target speaker, but our PGA just trains the adaptation model for 4.77 K steps with lower CER in the target domain and it controls the amount of involved parameters with 2 M.

4.2. Modular Analysis

To figure out the success of the proposed method, we perform a modular analysis by breaking down PGA and show the results in Table 2. The gating function f is first removed, the performance of EXP1 degrades on both the source domain and target domain, the amount of parameters reduces by 0.88 M, and training steps are not changed. Next, we remove the fine-tuning and just keep the adapter, in which only adapter parameters are trained on adaptation data. It is popular in NLP tasks that only fine-tune a small portion of the parameters while keeping most of the parameters shared across different tasks [36]. As

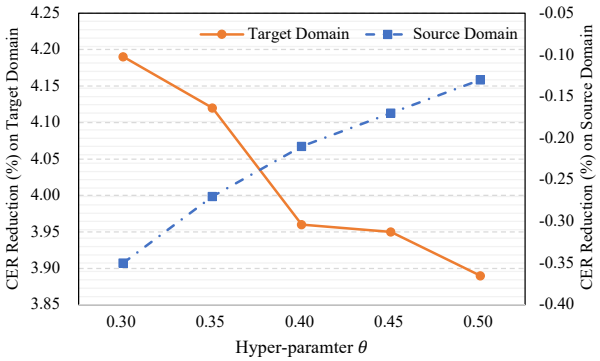


Figure 2: *The selection of hyper-parameter θ*

expected, EXP2 achieves excellent performance on the target domain but forgets some old knowledge leading to a 59.73% relative CER increase on the source domain compared to the baseline. In EXP3, we prune the adapter and just fine-tune the pre-trained encoder with adaptation data. EXP3 achieves limited performance improvement on the target domain and performs worse on the source domain. It can be seen that the gated fine-tuning method is crucial for avoiding catastrophic forgetting on the source domain, and PGA, which involves parallel adapters, demonstrates its powerful adaptation ability.

4.3. Hyperparameter Tuning

In this section, we explore the hyper-parameter θ denoted in section 2.2. When the value of θ increases, the gating score increases or decreases more slowly as α changes. Figure 2 shows the absolute CER reduction of PGA on the source domain and target domain, demonstrating that PGA benefits from the lower θ on the target domain, but performs better on the source domain as θ increases. To have a trade-off, we sum the CERs on the source domain and target domain and take the argument of the minimum 0.35 as the value of θ in our experiments.

4.4. Investigation of the data for Gated Fine-tuning

In the mentioned experimental results, all training data and 10-min target speaker data are used to fine-tune the whole ASR model with the gating function. A matter of concern is the storage of the amount of data for gated fine-tuning. Table 3 shows the results on different portions of training data for gated fine-tuning. When the amount of data for gated fine-tuning just be 1% of the training data, the CERs on the source domain and target domain both deteriorate slightly, and the training steps reduced from 4,770 to 72. That means the proposed PGA could achieve robust recognition including personalized recognition and generalized recognition, requiring only limited computational and storage resources.

5. Conclusion

In this paper, we propose an efficient speaker adaptation method termed personality-memory gated adaptation, which involves parallel adapters for target speaker adaptation and a gated fine-tuning mechanism based on personality similarity to avoid catastrophic forgetting. Experiments on Kespeech, MagicData-Zhengzhou, and MagicData-Sichuan demonstrate that, compared with the baseline, the proposed PGA significantly decreases the CER on the target domain and just brings a negligible CER increase on the source domain. Our method can achieve a trade-off between personality and generalizability with a small number of adaptation training steps and only 10-minute adaptation data.

6. References

[1] P. Bell, J. Fainberg, O. Klejch, J. Li, S. Renals, and P. Swietojanski, "Adaptation algorithms for neural network-based speech recognition: An overview," *IEEE Open Journal of Signal Processing*, vol. 2, pp. 33–66, 2020.

[2] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *TASLP*, vol. 19, no. 4, pp. 788–798, 2010.

[3] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *ICASSP*, 2018, pp. 5329–5333.

[4] S. Xue, O. Abdel-Hamid, H. Jiang, L. Dai, and Q. Liu, "Fast adaptation of deep neural network based on discriminant codes for speech recognition," *TASLP*, vol. 22, pp. 1713–1725, 2014.

[5] M. Delcroix, S. Watanabe, A. Ogawa, S. Karita, and T. Nakatani, "Auxiliary feature based adaptation of end-to-end asr systems." in *Interspeech*, vol. 2018, 2018, pp. 2444–2448.

[6] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *ASRU*, 2013, pp. 55–59.

[7] A. W. Senior and I. López-Moreno, "Improving DNN speaker independence with i-vector inputs," in *ICASSP*. IEEE, 2014, pp. 225–229.

[8] M. Zeineldeen, J. Xu, C. Lüscher, R. Schlüter, and H. Ney, "Improving the training recipe for a robust conformer-based hybrid model," in *INTERSPEECH*, 2022, pp. 1036–1040.

[9] L. Sarı, N. Moritz, T. Hori, and J. Le Roux, "Unsupervised speaker adaptation using attention-based speaker memory for end-to-end asr," in *ICASSP*, 2020, pp. 7384–7388.

[10] Z. Fan, J. Li, S. Zhou, and B. Xu, "Speaker-aware speech-transformer," in *ASRU*, 2019, pp. 222–229.

[11] Y. Zhao, C. Ni, C.-C. Leung, S. R. Joty, E. S. Chng, and B. Ma, "Speech transformer with speaker aware persistent memory," in *INTERSPEECH*, 2020, pp. 1261–1265.

[12] G. Wan, J. Pan, Q. Wang, J. Gao, and Z. Ye, "Speaker adaptive training for speech recognition based on attention-over-attention mechanism," in *INTERSPEECH*, 2020, pp. 1251–1255.

[13] K. Tomanek, V. Zayats, D. Padfield, K. Vaillancourt, and F. Biadsy, "Residual adapters for parameter-efficient asr adaptation to atypical and accented speech," in *EMNLP*, 2021, pp. 6751—-6760.

[14] Y. Huang, G. Ye, J. Li, and Y. Gong, "Rapid speaker adaptation for conformer transducer: Attention and bias are all you need," in *INTERSPEECH*, 2021, pp. 1309–1313.

[15] K. Yang, T.-Y. Hu, J.-H. R. Chang, H. S. Koppula, and O. Tuzel, "Text is all you need: Personalizing asr models using controllable speech synthesis," in *ICASSP*, 2023, pp. 1–5.

[16] P. Swietojanski, J. Li, and S. Renals, "Learning hidden unit contributions for unsupervised acoustic model adaptation," *TASLP*, vol. 24, no. 8, pp. 1450–1463, 2016.

[17] Z.-Q. Wang and D. Wang, "Unsupervised speaker adaptation of batch normalized acoustic models for robust asr," in *ICASSP*, 2017, pp. 4890–4894.

[18] X. Xie, X. Liu, T. Lee, and L. Wang, "Bayesian learning for deep neural network adaptation," *TASLP*, vol. 29, pp. 2096–2110, 2021.

[19] J. Deng, G. Li, X. Xie, Z. Jin, M. Cui, T. Wang, S. Hu, M. Geng, and X. Liu, "Factorised Speaker-environment Adaptive Training of Conformer Speech Recognition Systems," in *INTERSPEECH*, 2023, pp. 3342–3346.

[20] A. Bapna and O. Firat, "Simple, scalable adaptation for neural machine translation," in *EMNLP-IJCNLP*, 2019, pp. 1538–1548.

[21] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-efficient transfer learning for nlp," in *ICML*, 2019, pp. 2790–2799.

[22] Z. Hu, Y. Lan, L. Wang, W. Xu, E.-P. Lim, R. Ka-Wei Lee, L. Bing, and S. Poria, "Llm-adapters: An adapter family for parameter-efficient fine-tuning of large language models," in *EMNLP*, 2023, pp. 5254—-5276.

[23] Y. Li, A. Mehrish, R. Bhardwaj, N. Majumder, B. Cheng, S. Zhao, A. Zadeh, R. Mihalcea, and S. Poria, "Evaluating parameter-efficient transfer learning approaches on sure benchmark for speech understanding," in *ICASSP*, 2023, pp. 1–5.

[24] S. Radhakrishnan, C.-H. H. Yang, S. A. Khan, N. A. Kiani, D. Gomez-Cabrero, and J. N. Tegner, "A Parameter-Efficient Learning Approach to Arabic Dialect Identification with Pre-Trained General-Purpose Speech Model," in *INTERSPEECH*, 2023, pp. 1958–1962.

[25] G. Vanderreydt, A. Prasad, D. Khalil, S. Madikeri, K. Demuynck, and P. Motlicek, "Parameter-efficient tuning with adaptive bottlenecks for automatic speech recognition," in *ASRU*, 2023, pp. 1–7.

[26] Y. Gu, Z. Du, S. Zhang, Q. Chen, and J. Han, "Personality-aware Training based Speaker Adaptation for End-to-end Speech Recognition," in *INTERSPEECH*, 2023, pp. 1249–1253.

[27] S. Vander Eeckt and H. Van Hamme, "Using adapters to overcome catastrophic forgetting in end-to-end automatic speech recognition," in *ICASSP*, 2023, pp. 1–5.

[28] Z. Hu, L. Wang, Y. Lan, W. Xu, E.-P. Lim, L. Bing, X. Xu, S. Poria, and R. Lee, "LLM-adapters: An adapter family for parameter-efficient fine-tuning of large language models," in *EMNLP*, 2023, pp. 5254–5276.

[29] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," in *NIPS*, 2017, pp. 6306–6315.

[30] Z. Du, M. Lei, J. Han, and S. Zhang, "Pan: Phoneme-aware network for monaural speech enhancement," in *ICASSP*, 2020, pp. 6634–6638.

[31] Y. Ren, M. Lei, Z. Huang, S. Zhang, Q. Chen, Z. Yan, and Z. Zhao, "Prosospeech: Enhancing prosody with quantized vector pre-training in text-to-speech," in *ICASSP*, 2022, pp. 7577–7581.

[32] Z. Tang, D. Wang, Y. Xu, J. Sun, and et.al., "Kespeech: An open source speech dataset of mandarin and its eight subdialects," in *NeurIPS Datasets and Benchmarks*, 2021.

[33] M. Gibson, I. Karaulov, O. Zhelo, and F. Jurcicek, "A Personalised Speech Communication Application for Dysarthric Speakers," in *INTERSPEECH*, 2023, pp. 666–667.

[34] A. Gulati, J. Qin, C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented transformer for speech recognition," in *INTERSPEECH*, 2020, pp. 5036–5040.

[35] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "Espnet: End-to-end speech processing toolkit," in *INTERSPEECH*, 2018, pp. 2207–2211.

[36] Z. Fu, H. Yang, A. M. So, W. Lam, L. Bing, and N. Collier, "On the effectiveness of parameter-efficient fine-tuning," in *AAAI*. AAAI Press, 2023, pp. 12 799–12 807.