

Knowledge-Decoupled Functionally Invariant Path with Synthetic Personal Data for Personalized ASR

Yue Gu, Zhihao Du, Ying Shi, Jiqing Han, *Member, IEEE*, Yongjun He

Abstract—Fine-tuning generic ASR models with large-scale synthetic personal data can enhance the personalization of ASR models, but it introduces challenges in adapting to synthetic personal data without forgetting real knowledge, and in adapting to personal data without forgetting generic knowledge. Considering that the functionally invariant path (FIP) framework enables model adaptation while preserving prior knowledge, in this letter, we introduce FIP into synthetic-data-augmented personalized ASR models. However, the model still struggles to balance the learning of synthetic, personalized, and generic knowledge when applying FIP to train the model on all three types of data simultaneously. To decouple this learning process and further address the above two challenges, we integrate a gated parameter-isolation strategy into FIP and propose a knowledge-decoupled functionally invariant path (KDFIP) framework, which stores generic and personalized knowledge in separate modules and applies FIP to them sequentially. Specifically, KDFIP adapts the personalized module to synthetic and real personal data and the generic module to generic data. Both modules are updated along personalization-invariant paths, and their outputs are dynamically fused through a gating mechanism. With augmented synthetic data, KDFIP achieves a 29.38% relative character error rate reduction on target speakers and maintains comparable generalization performance to the unadapted ASR baseline.

Index Terms—personalized ASR, synthetic personal data, knowledge decoupling, functionally invariant path

I. INTRODUCTION

PERSONALIZED ASR models for individual speakers are essential in practice [1]. To enhance acoustic-level personalization, speaker adaptation techniques adapt generic ASR models to specific speakers by eliminating the mismatch in voice characteristics between training and testing.

Depending on whether speaker-specific adaptation data are incorporated, speaker adaptation methods are generally classified into embedding-based and model-based ones. The former typically incorporates speaker embeddings [2]–[5] or token embeddings [6], into the training of ASR models to obtain speaker-dependent ASR models [7]–[13]. Model-based methods fine-tune a generic ASR model [14], [15] or speaker-specific parameters [16]–[20] with speaker-specific data, enabling a better capture of the target speaker’s voice

characteristics. Thus, these types of methods generally outperform embedding-based approaches [21]. Recently, fine-tuning limited speaker-specific parameters instead of the entire model, such as parameter-efficient fine-tuning (PEFT) [22]–[25], holds a dominant position due to efficient adaptation. These methods follow a parameter-isolation [26] strategy that keeps non-target and target speaker parameters separate.

Although model-based speaker adaptation approaches have achieved certain success, the scarcity of speaker-specific data hinders further performance improvements, partly due to privacy concerns. Empirical evidence indicates that generating additional personal speech data with broader textual coverage is an effective strategy for data augmentation and improves personalized ASR [27]–[29]. Recently, large-scale zero-shot text-to-speech (TTS) models [30]–[34] have demonstrated human-level naturalness, expressiveness, and a diverse range of speaker profiles. These advances highlight their strong potential for data augmentation in low-resource ASR tasks [35], i.e., augmenting the real data of the target speaker with synthetic data to enhance personalized ASR models with respect to the speaker’s voice characteristics. However, hallucinations from large-scale TTS models [36] introduce phonetic or prosodic errors, potentially disturbing those model-based methods that rely on speaker-specific data. Moreover, when enhancing personalization with large-scale synthetic data, it is crucial for practical purposes to maintain generalization on non-target speakers, as considered in recent model-based adaptation methods [15], [20], [37].

The key to addressing the above challenges lies in enabling the model to learn synthetic personal data without forgetting real knowledge and to learn personal data without forgetting generic knowledge. The two learning tasks can be modeled as new data adaptations under functional invariance. Given that the functionally invariant path (FIP) framework [38] constructs a model updating path in the weight space that adapts to new data without impairing existing functionality, such as general recognition ability, we introduce FIP into personalized ASR tasks augmented with synthetic personal data.

When FIP is directly applied to personalized ASR, the model is trained on synthetic, real personal, and generic data, where large-scale synthetic data may compete with generic data, making it hard to balance different knowledge sources. To address this, we integrate the parameter-isolation strategy into FIP, storing generic and personalized knowledge in separate modules as in PEFT and applying FIP to them sequentially. To balance general representations from the generic module with speaker-specific information from the personalized module, we use the gating mechanism of our personality-

We sincerely thank Prof. Dong Wang (Tsinghua University) for valuable comments. This work was supported by NSFC under Grant 62376071. (Corresponding author: Zhihao Du, duzhihao.china@gmail.com; Jiqing Han, jqhan@hit.edu.cn.)

Yue Gu, Ying Shi, Jiqing Han, and Yongjun He are with the Research Center of Auditory Intelligence, School of Computer Science and Technology, Faculty of Computing, Harbin Institute of Technology, Harbin, China (e-mail: 427gy@sina.com; shiyings@hit.edu.cn; heyongjun@hit.edu.cn).

Zhihao Du is with the Speech Lab of Alibaba Group, Beijing, China.

memory gated adaptation (PGA) [20] to dynamically fuse their outputs. Accordingly, we propose knowledge-decoupled FIP (KDFIP) for synthetic-data-augmented personalized ASR models. After storing generic and personalized knowledge separately, KDFIP updates each module along the gated personalization-invariant paths, fully leveraging synthetic personal data while preserving generalization.

II. KDFIP WITH SYNTHETIC PERSONAL DATA

The schematic of the proposed KDFIP is shown in Fig. 1. Before detailing KDFIP, we first introduce synthetic data generation and the FIP framework to facilitate an intuitive understanding of the proposed method.

A. Data Augmentation with Zero-shot TTS Model

We augment the personal data with the zero-shot large-scale TTS model, CosyVoice 2.0. Our data augmentation process involves the following steps:

- 1) Randomly select an utterance (and its paired text segment y_{per}) in a given personal corpus as the reference speech and extract speech tokens μ_{per} , speaker embedding v , Mel-filter bank feature x_{per} . Then, randomly select a text segment y_{syn} from our internal multi-domain text database.
- 2) Construct the input sequence for text-to-token language model of CosyVoice 2.0 as “ $\{y_{\text{per}}, y_{\text{syn}}, \mu_{\text{per}}\}$ ” and generate the speech tokens μ_{syn} autoregressively for y_{syn} .
- 3) Feed the reference tokens μ_{per} , generated tokens μ_{syn} , reference feature x_{per} , and speaker embedding v to the conditional flow matching (CFM) model and generate speech features x_{syn} with ten iterations.
- 4) Using the vocoder to synthesize a waveform from x_{syn} .

To enhance the similarity between synthetic and real utterances, classifier-free guidance [39] is applied for both the LLM and CFM. Details are available in the repository¹.

B. FIP with Synthetic Personal Data for Personalized ASR

In FIP, a neural network is considered a smooth function $f(x; w)$ that maps an input vector x to an output vector y , where w denotes the trainable weights. FIP hypothesizes that the output $f(x; w)$ of a given neural network with a small weight perturbation dw can be approximated as follows:

$$f(x; w + dw) \approx f(x; w) + J_w dw \quad (1)$$

where J_w is the Jacobian of $f(x; w)$. Using this approximation, we can obtain the total difference between the outputs of two nearby networks with infinitesimal changes dw :

$$|f(x; w + dw) - f(x; w)|^2 = |\langle dw, dw \rangle_{r_w}|^2 \quad (2)$$

where $r_w = J_w(x)^T J_w(x)$ is the metric tensor that features the weight space as a Riemannian manifold. FIP hypothesizes that there is a functionally invariant path $\psi = \{w_k\}$ in the Riemannian manifold that minimizes the loss function L of a new task and output changes on the old task at the same time:

$$dw_k^* = \arg \min_{dw_k} \left(\left\langle dw_k, \frac{\partial L}{\partial w_k} \right\rangle + \beta \langle dw_k, dw_k \rangle_{r_{w_k}} \right) \quad (3)$$

¹<https://github.com/FunAudioLLM/CosyVoice>

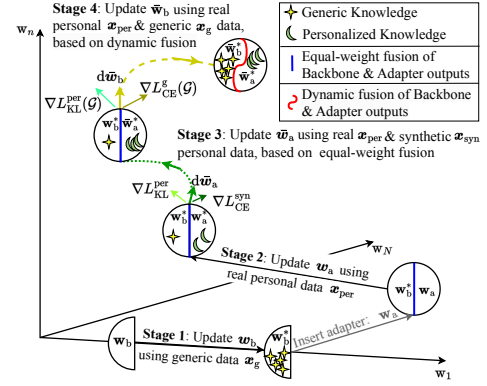


Fig. 1: Schematic of KDFIP construction in the weight space ($w_1, w_n \dots w_N$) of ASR models for sequential training on personal and generic data, where the generic module w_b and personalized module w_a correspond to $(w_1, \dots, w_n)$ and $(w_{n+1}, \dots, w_N)$, respectively. The spherical and hemispherical shapes represent sets of model parameters at each stage. ∇L and dw denote the gradient with respect to L and the perturbation in the weights w , respectively.

where β weighs the relative contribution of the two terms.

To improve personalization using synthetic personal data while avoiding forgetting generic and real knowledge, it is reasonable to introduce FIP to adapt the generic ASR model for the target speaker. Initially, given the generic ASR dataset $\{x_g, y_g\}$, we train the ASR model $f(x_g; w_b)$ to learn generic knowledge by minimizing the cross-entropy (CE) loss:

$$w_b^* = \arg \min_{w_b} \text{CE}(f(x_g; w_b), y_g) \quad (4)$$

where the parameters of the generic ASR model are denoted as w_b , which serves as the backbone model in KDFIP. This training process corresponds to “Stage 1” in Fig. 1. As in Eq. (3), the optimization problem of FIP can be addressed by iteratively minimizing a hybrid loss function $\mathcal{L}$ that consists of the classification loss for personalized ASR tasks and the distance in the output space of networks for generic ASR tasks:

$$\mathcal{L} = \text{CE}(f(\{x_{\text{per}}, x_{\text{syn}}\}; w_{\text{FIP}}), \{y_{\text{per}}, y_{\text{syn}}\}) + \beta \text{KL}(f(x_g; w_b^*), f(x_g; w_{\text{FIP}})) \quad (5)$$

where the trainable parameters w_{FIP} are initialized from w_b^* , and the KL divergence measures the distance between the outputs of $f(x_g; w_b^*)$ and $f(x_g; w_{\text{FIP}})$.

C. Knowledge-Decoupled FIP

Although FIP enables adaptation to real and synthetic personal data without catastrophic forgetting, it still struggles to simultaneously learn and balance personalized and generic knowledge. To tackle this issue, we propose knowledge-decoupled FIP (KDFIP), which adopts the parameter-isolation strategy and applies the FIP loss to independently learn personalized and generic knowledge. Following PEFT, we introduce additional adapters to each layer in the backbone encoder, with the hidden states of the backbone and adapter equally weighted. Given the limited amount of real personal data, augmenting it with synthetic personal data is a viable approach. Thus, the adapter is trained by minimizing the CE loss on both synthetic and real personal data:

$$\hat{w}_a = \arg \min_{w_a} \text{CE}(f(\{x_{\text{per}}, x_{\text{syn}}\}; \{w_b^*, w_a\}), \{y_{\text{per}}, y_{\text{syn}}\}) \quad (6)$$

where $f(\cdot; \{\mathbf{w}_b^*, \mathbf{w}_a\})$ denotes the ASR model equipped with adapters parameterized by $\mathbf{w}_a$. However, synthetic personal data may contain content or prosody errors due to language-model hallucinations, potentially impairing the training of adapters. The key challenge lies in learning from synthetic personal data without forgetting real personalized knowledge, which can be achieved by solving the **personalized functionality invariance** problem. In KDFIP, the adapter is first trained on personal data $\mathbf{x}_{\text{per}}$ to acquire real personalized knowledge as illustrated in “Stage 2” of Fig. 1:

$$\mathbf{w}_a^* = \arg \min_{\mathbf{w}_a} \text{CE}(f(\mathbf{x}_{\text{per}}; \{\mathbf{w}_b^*, \mathbf{w}_a\}), \mathbf{y}_{\text{per}}) \quad (7)$$

Then, the new adapter $\bar{\mathbf{w}}_a$, initialized from the original $\mathbf{w}_a^*$, is iteratively updated along personalization-invariant paths during “Stage 3” in Fig. 1, adapting to synthetic data without forgetting the real personalized knowledge contained in $\mathbf{w}_a^*$:

$$\begin{aligned} \bar{\mathbf{w}}_a^* &= \arg \min_{\bar{\mathbf{w}}_a} (L_{\text{CE}}^{\text{syn}} + L_{\text{KL}}^{\text{per}}) \\ &= \arg \min_{\bar{\mathbf{w}}_a} (\text{CE}(f(\mathbf{x}_{\text{syn}}; \{\mathbf{w}_b^*, \bar{\mathbf{w}}_a\}), \mathbf{y}_{\text{syn}}) \\ &\quad + \beta \text{KL}(f(\mathbf{x}_{\text{per}}; \{\mathbf{w}_b^*, \mathbf{w}_a^*\}), f(\mathbf{x}_{\text{per}}; \{\mathbf{w}_b^*, \bar{\mathbf{w}}_a\}))) \quad (8) \end{aligned}$$

The integration of personalized knowledge through the inserted adapters may influence the general recognition capability of the entire network. To restore generalization capabilities, it is necessary to dynamically fuse the learned knowledge. In PGA, a gating function $\mathcal{G}(\mathbf{x}, \mathcal{X}_{\text{per}})$ is utilized to control the proportion of adapter outputs $\mathbf{H}_a$ added to backbone outputs $\mathbf{H}_b$, based on the personality similarity between the input $\mathbf{x}$ and target speaker speeches $\mathcal{X}_{\text{per}} = \{\mathbf{x}_{\text{per}}\}$:

$$\mathbf{H} = \mathbf{H}_b + \mathcal{G}(\mathbf{x}, \mathcal{X}_{\text{per}}) \cdot \mathbf{H}_a \quad (9)$$

According to PGA, the backbone is retrained to minimize the CE loss on generic $\mathbf{x}_g$, personal $\mathbf{x}_{\text{per}}$, and synthetic data $\mathbf{x}_{\text{syn}}$:

$$\mathbf{w}_b^{\text{PGA}} = \arg \min_{\mathbf{w}_b} (\text{CE}(f(\{\mathbf{x}_g, \mathbf{x}_{\text{per}}, \mathbf{x}_{\text{syn}}\}; \{\mathbf{w}_b, \hat{\mathbf{w}}_a, \mathcal{G}\}), \{\mathbf{y}_g, \mathbf{y}_{\text{per}}, \mathbf{y}_{\text{syn}}\})) \quad (10)$$

where $\mathbf{w}_b$ is initialized from the backbone $\mathbf{w}_b^*$. However, the spectral discrepancy between synthetic and real personal data hinders the gating function from assigning high gating scores to synthetic data, indicating that such data should be excluded from Eq. (10). Nevertheless, optimizing the backbone by minimizing CE loss on real personal data $\mathbf{x}_{\text{per}}$ and generic data $\mathbf{x}_g$ is not applicable after “Stage 3”, as it may compromise the personalized knowledge acquired from synthetic data during “Stage 3”. We aim to recover the generalizability without forgetting the personalized knowledge acquired, which can be achieved by solving the problem of **personalized functionality invariance with the gating mechanism**. More precisely, KDFIP searches for an updating path of backbone parameters that preserves personalized functionality when the gating score is high and minimizes the CE loss on generic data when the gating score is low. Accordingly, new backbone parameters $\bar{\mathbf{w}}_b$, initialized from $\mathbf{w}_b^*$, are fine-tuned along gated personalization-invariance paths during “Stage 4” in Fig. 1:

TABLE I: Comparison of other speaker-adapted models: performance (CER, %), parameter number, and training steps.

Exps.	Model	Adaptation Data	Generic test set	Personal test set	# Parameter per speaker	Steps
Exp. 1	Base (Stage 1)		12.69	19.06	0	—
Exp. 2	SAT	N/A	12.70	18.00	0	—
Exp. 3	FT		13.14	17.82	0	—
Exp. 4	Adapter (Stage 2)	D_{per}	20.27	14.60	1.9M	20
Exp. 5	PGA		12.96	14.94	1.9M	—
Exp. 6	FT		20.81	13.79	0	39.9K
Exp. 7	Adapter	$D_{\text{per}} + D_{\text{syn}}$	27.47	14.57	1.9M	8.0K
Exp. 8	PGA		13.41	14.38	1.9M	13.1K
Exp. 9	FIP		13.93	14.17	0	39.9K
Exp. 10	Adapter-FIP (Stage 3)	$D_{\text{per}} + D_{\text{syn}}$	20.28	13.30	1.9M	2.4K
Exp. 11	KDFIP (Stage 4)		13.16	13.46	1.9M	15.5K

$$\begin{aligned} \bar{\mathbf{w}}_b^* &= \arg \min_{\bar{\mathbf{w}}_b} (L_{\text{CE}}^g + L_{\text{KL}}^{\text{per}}) \\ &= \arg \min_{\bar{\mathbf{w}}_b} (\text{CE}(f(\mathbf{x}_g; \{\bar{\mathbf{w}}_b, \bar{\mathbf{w}}_a^*, \mathcal{G}\}), \mathbf{y}_g) \\ &\quad + \beta \text{KL}(f(\mathbf{x}_{\text{per}}; \{\mathbf{w}_b^*, \bar{\mathbf{w}}_a^*\}), f(\mathbf{x}_{\text{per}}; \{\bar{\mathbf{w}}_b, \bar{\mathbf{w}}_a^*, \mathcal{G}\}))) \quad (11) \end{aligned}$$

Upon convergence, the final model, $f(\cdot; \{\bar{\mathbf{w}}_b^*, \bar{\mathbf{w}}_a^*, \mathcal{G}\})$, restores the generalizability of models on non-target speaker input while improving the personalization for the target speaker. In summary, KDFIP fully exploits synthetic data to augment personalized knowledge and avoids catastrophic forgetting of generic knowledge by integrating the FIP and PGA.

III. EXPERIMENTS

A. Experimental Setup

To evaluate the proposed KDFIP, we utilize three open-source transcribed audio datasets: KeSpeech [40], MagicData-Sichuan², and MagicData-Zhengzhou³. The text segments for synthetic data are selected from an internal dataset containing 130,000 sentences. The KeSpeech corpus contains 895 hours of utterances in Mandarin and its eight sub-dialects. According to the official data split, it is used as both the training dataset and the generic test set. As the unseen speaker corpora, MagicData-Sichuan and MagicData-Zhengzhou, which belong to Southwestern Mandarin and Zhongyuan Mandarin, respectively, are employed as the adaptation data (ten minutes for each speaker) and the personal test set (twenty minutes for each speaker). Consistent with PGA [20], this letter uses data from six speakers, such as “s_0001” and “z_0011”. The prefix “s” indicates Sichuan-accented Mandarin, while “z” denotes Zhengzhou-accented Mandarin. For a fair comparison, we use the same text dataset to generate nearly 100 hours of speech for each target speaker. We use the character error rate (CER) as the performance metric and also report the number of additional parameters per speaker and training steps to assess the model complexity and training cost of KDFIP.

The base (backbone) model, the gating function model, the adapter module, and the TTS model are pre-trained and publicly released [15], [20], [31], thus we use them directly in our experiments. The hyperparameter β is set to 0.01 in all experiments. The FIP ASR model is trained for 50 epochs. For the KDFIP setting, the adapter module is trained for three epochs with a learning rate of 0.0015, and the backbone is trained for three epochs with a learning rate of 5e-7.

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 II: Ablation study on “Stage 3” of the proposed KDFIP using synthetic data from target (s_1100) and non-target speakers (CER, %).

Adaptation Data	CER	Adaptation Data	CER
$D_{\text{per}}^{s_{1100}}$ + None	19.67	$D_{\text{per}}^{s_{1100}}$ + $D_{\text{syn}}^{s_{1100}}$	17.89
$D_{\text{per}}^{s_{1100}}$ + $D_{\text{syn}}^{s_{0001}}$	18.48	$D_{\text{per}}^{s_{1100}}$ + $D_{\text{syn}}^{s_{1106}}$	19.03
$D_{\text{per}}^{s_{1100}}$ + $D_{\text{syn}}^{z_{0011}}$	19.85	$D_{\text{per}}^{s_{1100}}$ + $D_{\text{syn}}^{z_{1196}}$	19.30

B. Experimental results

Table I presents the effectiveness of synthetic data augmentation and compares the proposed KDFIP with other speaker adaptation methods in terms of CER on generic and personal test sets. D_{per} means ten minutes real personal data and D_{syn} refers to 100 hours synthetic personal data. Notably, almost all models, except for SAT, share the same pre-trained backbone model. In speaker adaptive training (SAT), the speaker embedding is concatenated to the input of the conformer block to obtain speaker-dependent ASR models. Although this method does not require retraining, SAT ASR models depend on an external speaker embedding extractor and achieve only limited performance improvement on the personal test set.

As shown in the Table I, incorporating personal adaptation data into the ASR model training leads to better personalization performance. It is a simple and effective strategy that directly fine-tunes (FT) the pre-trained generic ASR model with adaptation data [14], [28]. Comparing the experiments Exp. 3 and Exp. 6, FT benefits more from the combination of synthetic and real personal data than from using only ten minutes of real personal data, demonstrating the effectiveness of synthetic personal data augmentation. In Exp. 7, the adapter is directly trained with CE loss on both synthetic and real personal data (Eq. (6)). Based on this, the PGA model in Exp. 8 is trained according to Eq. (10). Notably, the difference between Exp. 4 and Exp. 7, and between Exp. 5 and Exp. 8 lies in the use of synthetic data. In addition, the FIP model is trained following Eq. (5), while the Adapter-FIP model is trained during “Stage 3”. Comparing Exp. 4 and Exp. 7, we observe that the adapter fails to benefit from the synthetic data, as Exp. 7 shows degraded performance on the generic dataset. This is possibly due to the adapter being sensitive to phonetic errors introduced by hallucinations of the TTS model. As a result, although the PGA model (Exp. 8) maintains generalizability when synthetic data is used, it performs worse than FT (Exp. 6) and KDFIP on personal test sets.

The FIP model struggles to learn from the generic data, real and synthetic personal data simultaneously. Without knowledge decoupling, it becomes difficult for FIP to balance multiple types of knowledge, leading to suboptimal performance on both generic and personal test sets. Compared with the base model, KDFIP achieves the highest relative improvement of 29.38% among the aforementioned methods on personal test sets, with only a slight deterioration in generalizability. In addition, KDFIP requires only a limited number of training steps for adaptation.

C. Ablation study

To examine the effect of the acoustic information from target speakers, we conduct an ablation experiment by combining ten minutes of real personal data from speaker “s_1100” with

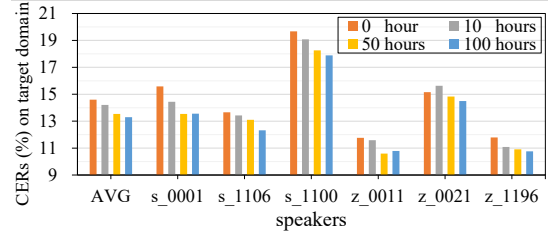


Fig. 2: Ablation on the duration of synthetic personal data in “Stage 3” of KDFIP.

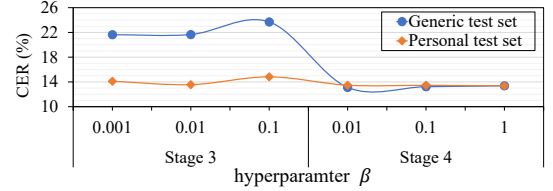


Fig. 3: Interpolated hyperparameter tuning of KDFIP.

100 hours of synthetic personal data from other speakers in “Stage 3” of KDFIP, based on the “Stage 2” model of speaker “s_1100”. As shown in Table II, although the synthetic data from other speakers shares the same textual context as the target speaker, the ASR model performs better on the target speaker when the corresponding synthetic data is involved. We believe this improvement is related to the powerful capabilities of large TTS models.

Fig. 2 illustrates the influence of the synthetic personal data duration in “Stage 3”. Overall, increasing the amount of synthetic personal data leads to performance improvements. However, there is no clear difference between 50 and 100 hours of adaptation data for some speakers, such as “s_0001” and “z_1196”. This may be due to the underrepresentation of the target speaker’s voice characteristics in the limited ten-minute real personal dataset, causing the TTS model to generate data with insufficient speaker profiles.

D. Hyperparameter Tuning

The interpolated hyperparameter β , denoted in the method section, is tuned to assist model training. Fig. 3 shows the CERs on both generic and personal test sets of “Stage 3” and “Stage 4” as β varies. The figure indicates that the ASR model is robust to the change of β during “Stage 4”, while it is sensible to larger β , such as 0.1, during “Stage 3”. In this letter, β is set to 0.01 for all experiments.

IV. CONCLUSION

It is difficult for personalized ASR models to learn and balance the synthetic knowledge, real personalized knowledge, and generic knowledge when amounts of synthetic personal data are involved in the model training. In this letter, we propose KDFIP to decouple the learning of three types of knowledge into personalization and generalizability adaptations, significantly improving personalized ASR without compromising generalizability. Experimental results on three open-source datasets demonstrate the effectiveness of KDFIP and synthetic personal data augmentation. Additionally, the ablation study indicates that the target speaker’s voice characteristics are a critical factor in data augmentation using large-scale TTS models.

REFERENCES

- [1] M. Lee, J. Mo, J. Kang, J. Son, and J. Chang, “Bayesian language model adaptation for personalized speech recognition,” *SPL*, vol. 32, pp. 1620–1624, 2025.
- [2] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, “Front-end factor analysis for speaker verification,” *TASLP*, vol. 19, pp. 788–798, 2011.
- [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*, 2018, pp. 2444–2448.
- [6] S. Li, D. Wei, H. Shang, J. Guo, Z. Li *et al.*, “Speaker-smoothed knn speaker adaptation for end-to-end asr,” in *INTERSPEECH*, 2024, pp. 2390–2394.
- [7] 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.
- [8] A. W. Senior and I. López-Moreno, “Improving DNN speaker independence with i-vector inputs,” in *ICASSP*, 2014, pp. 225–229.
- [9] 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.
- [10] L. Sari, 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.
- [11] Z. Fan, J. Li, S. Zhou, and B. Xu, “Speaker-aware speech-transformer,” in *ASRU*, 2019, pp. 222–229.
- [12] 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.
- [13] 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.
- [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] 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.
- [16] P. Swietojanski, J. Li, and S. Renals, “Learning hidden unit contributions for unsupervised acoustic model adaptation,” *TASLP*, vol. 24, 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] Y. Gu, Z. Du, S. Zhang, Jiqing Han, and Y. He, “Personality-memory gated adaptation: An efficient speaker adaptation for personalized end-to-end automatic speech recognition,” in *INTERSPEECH*, 2024, pp. 2870–2874.
- [21] 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.
- [22] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe *et al.*, “Parameter-efficient transfer learning for nlp,” in *ICML*, 2019, pp. 2790–2799.
- [23] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, “Lora: Low-rank adaptation of large language models,” in *ICLR*, 2022.
- [24] 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.
- [25] Y. Li, A. Mehrish, R. Bhardwaj, N. Majumder, B. Cheng *et al.*, “Evaluating parameter-efficient transfer learning approaches on sure benchmark for speech understanding,” in *ICASSP*, 2023, pp. 1–5.
- [26] Z. Wang, Y. Liu, T. Ji, X. Wang, Y. Wu, C. Jiang, Y. Chao, Z. Han, L. Wang, X. Shao, and W. Zeng, “Rehearsal-free continual language learning via efficient parameter isolation,” in *ACL (1)*. Association for Computational Linguistics, 2023, pp. 10933–10946.
- [27] Y. Huang, L. He, W. Wei, W. Gale, J. Li, and Y. Gong, “Using personalized speech synthesis and neural language generator for rapid speaker adaptation,” in *ICASSP*, 2020, pp. 7399–7403.
- [28] 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.
- [29] D. Kim, J. Lee, and J. Chang, “Text-only unsupervised domain adaptation for neural transducer-based ASR personalization using synthesized data,” in *ICASSP*, 2024, pp. 11 131–11 135.
- [30] Z. Du, Q. Chen, S. Zhang, K. Hu, H. Lu *et al.*, “Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens,” *arXiv preprint arXiv:2407.05407*, 2024.
- [31] Z. Du, Y. Wang, Q. Chen, X. Shi, X. Lv *et al.*, “Cosyvoice 2: Scalable streaming speech synthesis with large language models,” *arXiv preprint arXiv:2412.10117*, 2024.
- [32] Z. Du, C. Gao, Y. Wang, F. Yu, T. Zhao *et al.*, “Cosyvoice 3: Towards in-the-wild speech generation via scaling-up and post-training,” *arXiv preprint arXiv:2505.17589*, 2025.
- [33] P. Anastassiou, J. Chen, J. Chen, Y. Chen, Z. Chen *et al.*, “Seed-tts: A family of high-quality versatile speech generation models,” *arXiv preprint arXiv:2406.02430*, 2024.
- [34] M. Le, A. Vyas, B. Shi, B. Karrer, L. Sari *et al.*, “Voicebox: Text-guided multilingual universal speech generation at scale,” *Advances in neural information processing systems*, vol. 36, pp. 14 005–14 034, 2023.
- [35] G. Yang, F. Yu, Z. Ma, Z. Du, Z. Gao, S. Zhang, and X. Chen, “Enhancing low-resource asr through versatile tts: Bridging the data gap,” in *ICASSP*, 2025, pp. 1–5.
- [36] C. Liu, M. Fang, P. Zhang, W. Zhou, J. Gao, and J. Han, “Mitigating hallucinations in lm-based tts models via distribution alignment using gflownets,” in *EMNLP*, 2025, accepted for publication. [Online]. Available: <https://arxiv.org/abs/2508.15442>
- [37] S. Vander Eeck and H. Van Hamme, “Using adapters to overcome catastrophic forgetting in end-to-end automatic speech recognition,” in *ICASSP*, 2023, pp. 1–5.
- [38] G. Raghavan, B. Tharwat, S. N. Hari, D. Satani, R. Liu, and M. Thomson, “Engineering flexible machine learning systems by traversing functionally invariant paths,” *Nature Machine Intelligence*, vol. 6, pp. 1179–1196, 2024.
- [39] J. Ho and T. Salimans, “Classifier-free diffusion guidance,” *arXiv preprint arXiv:2207.12598*, 2022.
- [40] Z. Tang, D. Wang, Y. Xu, J. Sun, and et.al., “Ksespeech: An open source speech dataset of mandarin and its eight subdialects,” in *NeurIPS Datasets and Benchmarks*, 2021.