THE NPU-CHINATELECOM SYSTEM FOR SLT2024 SOURCE SPEAKER TRACING CHALLENGE

Qing Wang[1,2], *Hongmei Guo*[2,3], *Jian Kang*[2], *Mengjie Du*[2], *Jie Li*[2], *Xiao-Lei Zhang*[2,3], *Lei Xie*[1]

[1]Audio, Speech and Language Processing Group (ASLP@NPU), School of Computer Science,
Northwestern Polytechnical University, Xi'an, China
[2]Institute of Artificial Intelligence (TeleAI), China Telecom, China
[3]School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an, China

ABSTRACT

The Source Speaker Tracing Challenge (SSTC) in IEEE SLT2024 is designed to identify the information of the source speaker in manipulated speech signals. Specifically, SLT2024 SSTC focuses on source speaker verification against voice conversion to determine whether two converted speech samples originate from the same source speaker. This paper introduces our proposed source speaker tracing system submitted to the challenge. To learn a more source-speaker-related representation, our system utilizes a source speaker contrastive loss to learn the latent source speaker information in the converted speech, enhancing source speaker tracing. Experiments demonstrate that our system achieves an EER of 16.788% on the challenge test set, securing 2nd place in the challenge.

Index Terms— Source Speaker Tracing Challenge, source speaker verification, voice conversion

1. INTRODUCTION

Speaker verification (SV) is a critical biometric authentication technology used in various daily applications, such as identity verification, personalized devices, and financial transactions. Protecting the security of SV systems against a range of threats, including spoofing and adversarial attacks [1–3], has become increasingly important.

As a typical type of spoof attack, voice conversion (VC) [4, 5] is a technology that modifies the characteristics of a person's speech (source speaker) to make it sound like another person's (target speaker) voice while retaining the original content. To address the threat of VC in speaker verification, the SLT2024 Source Speaker Tracing Challenge (SSTC)[1] has introduced the task of source speaker verification against voice conversion. This task aims to determine whether the source speakers of converted speech are the same, as converted speech retains certain aspects of the source speaker's speaking style [6].

This paper aims to contribute to the development of source speaker tracing and serves as a submission to the SLT2024 SSTC. To make the latent source speaker information more revealed in the embedding of converted speech, we propose source speaker contrastive learning. This system employs source speaker contrastive loss to enhance model training, enabling the generation of more discriminative embeddings for source speakers. Our approach integrates a speaker embedding extractor trained with converted speech data, learning to capture the source speaker's characteristics that remain after conversion. Experimental results on the SLT2024 SSTC datasets demonstrate a significant performance improvement compared to baseline systems.

2. SYSTEM DESCRIPTION

In this section, we detail the systems we submitted to the challenge, which are the baselines including MFA-Conformer and MFA-Conformer with adaptor, and the proposed source speaker contrastive learning method.

2.1. Speaker Embedding Clustering

The first official baseline is based on MFA-Conformer [7, 8]. The speaker embedding network is designed to create a distinct embedding space where utterances from the same speaker are grouped, while those from different speakers are separated. Voice conversion models alter the source speech to resemble the target speaker's voice, aiming to trick the speaker verification system. Consequently, converted speech often lies within the target speaker's subspace. Source speaker identification, on the other hand, seeks to map the converted speech back to the source speaker's subspace.

Starting with a source speech dataset $\mathcal{D}_s = \{s_i\}$ and a target speech dataset $\mathcal{D}_t = \{t_j\}$, a voice conversion model modifies the voice of each source speech s_i to sound like the voice of a target speech t_j, resulting in a converted speech dataset $\mathcal{D}_c = \{x_{s_i \to t_j} \mid s_i \in \mathcal{D}_s, t_j \in \mathcal{D}_t\}$. To train the speaker embedding network, the datasets $\mathcal{D}_s$, $\mathcal{D}_t$, and the

[1] https://sstc-challenge.github.io/

converted speech datasets $\mathcal{D}_{c,k}$ generated by the voice conversion algorithms C_k (for $k = 1, \ldots, K$) are combined into the training dataset $\mathcal{D} = \mathcal{D}_s \cup \mathcal{D}_t \cup \mathcal{D}_{c,1} \cup \cdots \cup \mathcal{D}_{c,K}$. During the training process, the label assigned to each converted speech $x_{s_i \to t_j}$ is the speaker identity of the source speech s_i. The speaker embedding network is thus optimized to capture the source speaker's characteristics from the converted speech while maintaining a discriminative embedding space.

2.2. Adapter-based Multi-Task Learning

To further improve the ability to distinguish the converted speech by different kinds of VC methods, the second official baseline is based on MFA-Conformer with adaptor [9]. The integration of source speaker verification and conversion method recognition tasks acknowledges their inherent connection. Despite their distinct objectives, these tasks share interrelated information present in converted speech samples. Multi-task learning provides a unified approach to tackle both objectives concurrently. By utilizing task-specific parameter sets embedded within adapter modules in the model architecture, shared learning across tasks is facilitated, enabling the model to simultaneously identify both the source speaker and the specific conversion method.

During training, the model learns to distinguish between known and unseen conversion methods. Utterances from the same conversion method tend to cluster together, whereas those from different methods exhibit greater dispersion. This clustering effect aids in recognizing known methods included in the training set. However, identifying unseen methods poses a challenge due to the impracticality of encompassing all potential techniques in the training data. Nevertheless, the model effectively utilizes similarities in Euclidean distances between audio samples to distinguish between known and unseen conversion methods. This approach equips the model to robustly handle both types of methods during inference, thereby enhancing its practical utility in real-world applications.

2.3. Source Speaker Contrastive Learning

In addition to the officially provided baseline mentioned above, we propose a source speaker contrastive learning for the source speaker tracing task. The training procedure of our proposed system is outlined in Fig 1. Once the speaker embedding extractor is trained, it remains fixed to extract embeddings from source speech, which are utilized as positive and negative samples to compute the speaker contrastive loss. Subsequently, the speaker embedding extractor network continues to be trained using converted speech, and the representation from a fully connected layer is employed for computing the speaker contrastive loss.

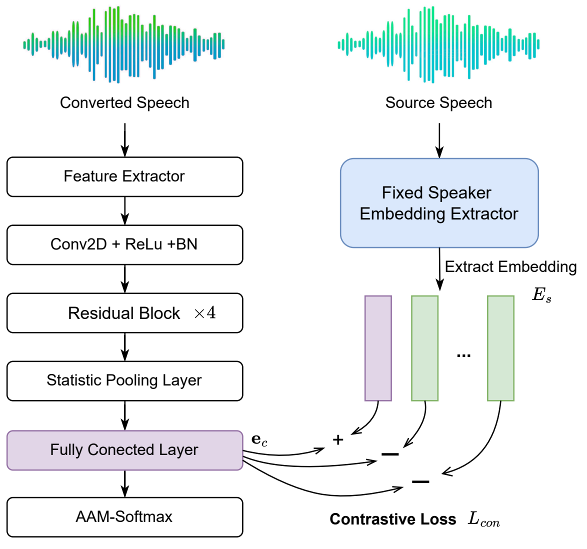

Fig. 1. Overview of the proposed source speaker contrastive learning system. $\mathbf{e}_c$ and E_s refer to the representation of the embedding extraction layer and a set of embeddings of source speech, respectively. Here, $+$ and $-$ represent positive and negative sample pairs, respectively, which are used to compute the speaker contrastive loss $\mathcal{L}_{\text{Con}}$.

The source speaker contrastive loss is adopted to identify the true source speaker embedding within K distractors of speaker embeddings so that the embedding extractor can learn the potentially possessing source speaker information in the converted speech. Given a converted speaker embedding $\mathbf{e}_c$ and a set of source speaker embeddings $E_s = \{\mathbf{e}_s^1, \ldots, \mathbf{e}_s^K\}$, the contrastive loss can be calculated as:

$$\mathcal{L}_{\text{Con}} = -\log \frac{\exp(\cos(\mathbf{e}_c, \mathbf{e}_k)/\tau)}{\sum_{e_s \sim E_s} \exp(\cos(\mathbf{e}_c, \mathbf{e}_s)/\tau)}, \tag{1}$$

where $\cos(\cdot)$ represents the cosine similarity between two vectors and τ is a temperature hyper-parameter. Here, E_s is a set comprising all candidate source speaker embeddings, where the positive sample $\mathbf{e}_k$ belongs to the same speaker as converted embedding $\mathbf{e}_c$ and others are negative samples from different speakers.

The final loss function of our source speaker contrastive learning model is given by:

$$\mathcal{L} = \mathcal{L}_{\text{AAM}} + \alpha \mathcal{L}_{\text{Con}}, \tag{2}$$

where $\mathcal{L}_{\text{AAM}}$ is the loss function of the speaker embedding extractor, α refers to an interpolation factor that scales the speaker contrastive loss. We sample $K = 5$ negative samples during the training and set $\alpha = 1$ to ensure that the two losses are of a similar magnitude.

Table 1. The EER Results (%) of Source Speaker Verification on Development Sets.

Method	**Dev-1**	**Dev-2**	**Dev-3**	**Dev-4**	**Dev-5**	**Dev-6**	**Dev-7**	**Dev-8**	**Dev-9**	**Dev-10**	**Dev-11**	**Dev-12**
MFA-Conformer	10.781	10.191	9.034	9.232	8.699	12.984	27.137	24.061	33.360	45.674	19.734	21.668
+**Adapter**	9.535	8.971	8.290	8.013	7.861	12.919	30.129	27.999	33.189	45.060	20.599	21.564
ResNet293	9.327	9.918	7.876	7.852	7.119	10.752	24.245	21.934	29.310	40.158	19.017	20.132
$+\mathcal{L}_{\textbf{Con}}$	8.305	9.629	7.634	7.534	6.805	10.051	23.308	20.165	27.893	39.276	18.032	18.984

Table 2. The EER Results (%) of Source Speaker Verification on Test Set.

Method	EER
MFA-Conformer	20.374
+**Adapter**	20.046
ResNet293	18.626
$+\mathcal{L}_{\textbf{Con}}$	16.788

3. EXPERIMENTS SETUP AND RESULT

3.1. Dataset

For model training, our source speaker tracing systems are trained on the train-clean section of Librispeech [10] and Voxceleb2 development set [11], which are used as the source and target speakers for voice conversion, respectively. In addition, the training and development sets [9] are also used for training, fine-tuning and evaluate the models.

3.2. Setup

The setups of systems we submitted to the SSTC2024 are as follows.

MFA-Conformer with adaptor[2]: As described in Sections 2.1 and 2.2, we use MFA-Conformer [7, 8] for model training. For the front-end processing, we extract 80 dimensional log Mel-filterbank energies using a frame length of 25ms and a hop size of 10ms. These features are mean normalized before being fed into the deep speaker network. As mentioned in Section 2.1's back-end part differs slightly from Section 2.2. In Speaker Embedding Clustering, an Encoder-Decoder structure is used. In the Encoder, a series of 8 projection layers map the input features to another 176-dimensional hidden representation. This feature representation undergoes Layer Normalization. After pooling the features using attentive statistics pooling, batch normalization is applied again. Finally, a linear layer with an output dimension of 256 maps the features to the final output dimension, and Dropout is applied to prevent overfitting. In adapter-based multi-task Learning, the MFA-conformer model is modified by inserting an adapter after each conformer block within the MFA-Conformer architecture.

ResNet293 with speaker contrastive loss: The model is training in three phases. For Phase 1: we follow the standard 293-layer ResNet architecture [12] to train the ResNet293 system. The additive angular margin (AAM) loss [13] is used as the training objective. The scale and margin in the AAM loss are set to 32 and 0.2 respectively. The ResNet293 is trained using the Librispeech train-clean set. In Phase 2, the training set of SSTC 2024 (converted speech) and Librispeech data (source speech) is used to fine-tune the model. Then, the converted speech is used to train the model with contrastive loss and AAM loss, with the source speech embedding (extracted by the Phase 1 model). The hyper-parameters are set as $K = 5$ and $\alpha = 1$.

3.3. Evaluation Results

SSTC 2024 uses the Equal Error Rate (EER) metric to evaluate the performance of the systems. For each pair of two converted speech utterances in the development and test sets, the cosine similarity is computed, and the decision on same-source-speaker vs. different-source-speakers is made by a threshold.

Table 1 presents the EER results of the development set. For further applying the in-domain information, Dev sets are also added to fintune the models. The EER results on SSTC test sets of different methods are shown in Table 2. Our proposed source speaker contrastive learning method achieves an EER of 16.788% with a 3.825% improvement over the official baseline, which demonstrates the latent source speaker information learned by the source speaker contrastive loss is beneficial to the source speaker tracing.

4. CONCLUSION

In this paper, we present our NPU-ChinaTelecom system for the SLT2024 Source Speaker Tracing Challenge (SSTC), focusing on enhancing the source speaker tracing against voice conversion. Our approach leverages source speaker

[2] https://github.com/SSTC-Challenge/SSTC2024_baseline_system

contrastive learning to effectively capture and utilize source speaker information embedded in converted speech. Evaluation of the SSTC 2024 dataset indicates that our proposed system achieves a 16.788% EER on the test set, ranking 2nd place in the challenge.

5. REFERENCES

[1] Zhizheng Wu, Junichi Yamagishi, Tomi Kinnunen, Cemal Hanilçi, Mohammed Sahidullah, Aleksandr Sizov, Nicholas Evans, Massimiliano Todisco, and Hector Delgado, "ASVspoof: the automatic speaker verification spoofing and countermeasures challenge," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 4, pp. 588–604, 2017.

[2] Massimiliano Todisco, Xin Wang, Ville Vestman, Md Sahidullah, Héctor Delgado, Andreas Nautsch, Junichi Yamagishi, Nicholas Evans, Tomi Kinnunen, and Kong Aik Lee, "ASVspoof 2019: Future horizons in spoofed and fake audio detection," *arXiv preprint arXiv:1904.05441*, 2019.

[3] Qing Wang, Pengcheng Guo, and Lei Xie, "Inaudible adversarial perturbations for targeted attack in speaker recognition," *Proc. INTERSPEECH*, pp. 4228–4232, 2020.

[4] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[5] Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu, "Fastspeech: Fast, robust and controllable text to speech," *Proc. NeurIPS*, vol. 32, 2019.

[6] Danwei Cai, Zexin Cai, and Ming Li, "Identifying source speakers for voice conversion based spoofing attacks on speaker verification systems," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[7] Yang Zhang, Zhiqiang Lv, Haibin Wu, Shanshan Zhang, Pengfei Hu, Zhiyong Wu, Hung-yi Lee, and Helen Meng, "Mfa-conformer: Multi-scale feature aggregation conformer for automatic speaker verification," *arXiv preprint arXiv:2203.15249*, 2022.

[8] Danwei Cai and Ming Li, "Leveraging asr pretrained conformers for speaker verification through transfer learning and knowledge distillation," *arXiv preprint arXiv:2309.03019*, 2023.

[9] Ze Li, Yuke Lin, Tian Yao, Hongbin Suo, and Ming Li, "The database and benchmark for source speaker verification against voice conversion," *arXiv preprint arXiv:2406.04951*, 2024.

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

[11] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[12] Zhengyang Chen, Bei Liu, Bing Han, Leying Zhang, and Yanmin Qian, "The sjtu x-lance lab system for cnsrc 2022," *arXiv preprint arXiv:2206.11699*, 2022.

[13] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4690–4699.