Interspeech 2024
1-5 September 2024, Kos, Greece

Centroid Estimation with Transformer-Based Speaker Embedder for Robust Target Speaker Extraction

Woon-Haeng Heo, Joongyu Maeng, Yoseb Kang, Namhyun Cho

Audio AI Lab., NC Research, NCSOFT Corp., Republic of Korea
{luckyheo, mkyu, ky0, cnh2769}@ncsoft.com

Abstract

Target speaker extraction (TSE) is a technique for separating the target speaker from mixed speech using speaker embedding. However, speaker embeddings may contain, in addition to speaker information, text dependent information and environmental information, such as noise, microphone characteristics, and reverberation, which can decrease TSE performance, especially when the enrollment and target utterances are in different environments. To address this issue, we propose a Transformer-based embedder for centroid estimation, and a role division training method to enhance the training stability of the TSE separator. This embedder estimates the speaker centroid from the enrollment utterance, aiding the separator in extracting the target speaker. The proposed methods considerably improve speech quality and speech recognition performance compared to the baseline.

Index Terms: target speaker extraction, Transformer, robust speaker embedder, speech recognition

1. Introduction

The "cocktail party problem" has long been challenging in speech processing, and recent studies have made significant strides in addressing it through speech enhancement and separation techniques [1, 2, 3, 4]. However, distortion can negatively impact automatic speech recognition (ASR) performance when a speech enhancement module is used to remove background noise (non-speech noise) prior to ASR [5, 6]. An alternative approach that mitigates this issue is to train a robust ASR model by mixing background noise with the speech signal. Although background noise can be easily learned due to its distinct characteristics from speech, the processing of overlapped speech (speech noise) is more complex. The two main approaches for processing overlapped speech are speech separation and target speaker extraction (TSE) [7]. In the speech separation task, the permutation problem makes it impossible to determine the speakers of separated speech [1, 2, 3]. However, TSE can avoid this problem if there is a speaker embedding vector of the target speaker [7].

The pretrained TSE embedder estimates speaker embeddings from the enrollment utterance, which are used alongside mixed speech as input for the separator to extract the target speaker [7, 8]. There are various studies on different approaches to the TSE task. In several recent studies, an embedder and separator were jointly trained without using a pretrained embedder [9, 10]. Additionally, the speech separator models are used, which have shown good performance in the speech separation task [8, 11]. In another approach, research by Sato et al. [12] utilized a collection of worst enrollment cases for training to address performance degradation in the TSE task due to the noise in enrollment utterances or differences in speech style between the target utterance and enrollment utterances. On the other hand, Mun et al. [13] uses speaker centroids, improving the TSE performance by incorporating the verification loss between the target speaker centroid and the embedding vector of the estimated target speech into the training process.

In this study, we conducted several preliminary experiments and then obtained two interesting results. First, numerous failures were observed when the environments of the enrollment and target utterances differed. Second, we experimented with alternative embeddings [14, 15] instead of the d-vector [16] used in previous experiments. Surprisingly, we found that TSE performance decreased despite an increase in speaker validation performance. Based on the aforementioned results, we opted to utilize the centroid (or prototype) of the d-vector as the speaker embedding. This centroid represents a robust speaker embedding, characterized by being far from multiple negative factors, such as text, noise, microphone characteristics, and reverberation. Therefore, we added the Transformer architecture [17] that was used in a previous speaker embedding adaptation study [18] to the embedder in order to estimate the centroid.

We propose a Transformer-based speaker embedder to estimate a centroid for robust TSE, including using a role division training (RDT) method. The Transformer-based speaker embedder, comprising multi-head attention (MHA) and attentive statistics pooling (ASP) [19], processes the frame-by-frame d-vector embedding of the enrollment utterance. It first transforms d-vector embedding via MHA, then aggregates them into a single embedding using ASP to derive the centroid. The proposed architecture introduces additional parameters solely to the embedder, which can be pre-computed and stored independently of the separator. Consequently, this design provides the advantage of not affecting the real-time factor (RTF) of the existing structure. Furthermore, our training strategy entails initially training the embedder and separator separately, followed by a joint learning phase. This approach aims to mitigate performance degradation arising from divergent convergence rates between the two components.

The purpose of RDT is to enhance the training stability of the TSE separator. The RDT entails the selective use of batches characterized by moderately low signal loss. In the TSE, the role of the embedder is to identify the target speaker, whereas the separator aims to extract the target speaker with high quality. A large signal loss is indicative of an output from an interference speaker, attributing to inaccuracies in the embedder.

We conducted several experiments to evaluate the proposed method, and our results showed improvements in speech quality, failure rate, and overall ASR performance compared to the baseline. Specifically, the failure rate improved significantly when the environments of the enrollment and target utterances

10.21437/Interspeech.2024-1560

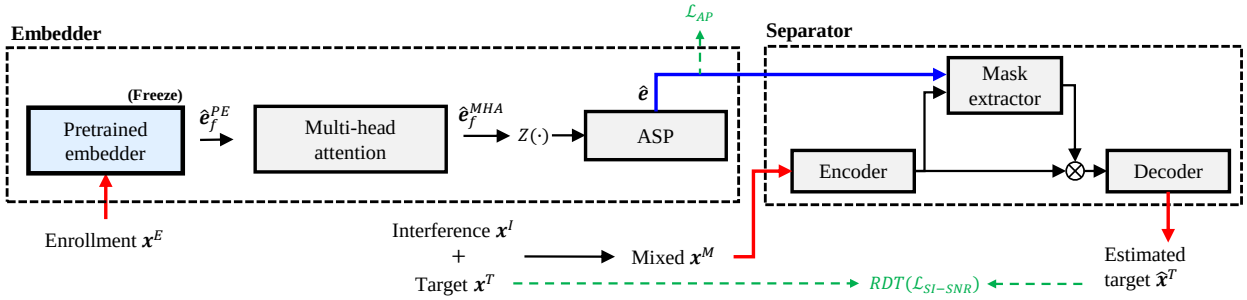

Figure 1: *Proposed architecture. The red solid arrows (representing the inputs and output of TSE), the blue solid arrows (representing the transfer of speaker embedding), and the black solid arrows illustrate the inference process. The green dashed arrows represent the losses or training method used in training.*

differed. As a short summary, the main contributions of this work include:

1. We propose a Transformer-based speaker embedder to estimate a centroid that is distant from multiple negative factors such as text, noise, microphone characteristics, and reverberation.
2. We propose a role division training method to enhance the training stability of the separator.
3. We evaluate the proposed methods using various metrics, such as speech quality, failure rate, and ASR performance. The proposed methods improve overall performance while maintaining the RTF of the existing TSE.

2. Proposed methods

2.1. Model architecture

The model architecture depicted in Figure 1 comprises an embedder $Emb(\cdot)$ and a separator $Sep(\cdot)$. The embedder utilizes MHA, which is a Transformer encoder. The architecture of the embedder is as follows: pretrained embedder (PE), MHA, and ASP. The pretrained embedder is a speaker embedding model trained using a generalized end-to-end (GE2E) loss function [16], called d-vector. The separator has an X-TasNet architecture [8]. The enrollment utterance $\mathbf{x}^E$ and mixed speech $\mathbf{x}^M$ are fed as inputs into the proposed architecture $M(\cdot)$. The process of extracting the target speech $\mathbf{x}^T$ can be expressed using the following equation:

$$\hat{\mathbf{x}}^T = M\left(\mathbf{x}^M, \mathbf{x}^E; \theta_{emb}, \theta_{sep}\right) = Sep\left(\mathbf{x}^M, \hat{\mathbf{e}}; \theta_{sep}\right), \quad (1)$$

$$\hat{\mathbf{e}} = Emb\left(\mathbf{x}^E, \theta_{emb}\right) = ASP\left(Z\left(MHA\left(PE\left(\mathbf{x}^E\right)\right)\right)\right), \quad (2)$$

where $Z(\cdot)$ represents L2-normalization, and θ_{emb}, θ_{sep} are the parameters of the embedder and separator, respectively. The output of the embedder is denoted by $\hat{\mathbf{e}}$ and the embedding variable $\mathbf{e}$ is denoted as $\mathbf{e} \in \mathbf{R}^D$. Mixed speech $\mathbf{x}^M$ is generated by mixing target speech $\mathbf{x}^T$ and interference speech $\mathbf{x}^I$ having a random signal-to-interference ratio (SIR). To provide a more detailed expression of the feature flow in the pretrained embedder and MHA, we use the following equation:

$$\hat{\mathbf{e}}_f^{PE} = PE\left(\mathbf{x}^E\right), \quad (3)$$

$$\hat{\mathbf{e}}_f^{MHA} = MHA\left(PE\left(\mathbf{x}^E\right)\right), \quad (4)$$

where f is the frame. The MHA transforms frame-by-frame d-vectors $\hat{\mathbf{e}}_f^{PE}$ into frame-by-frame centroids $\hat{\mathbf{e}}_f^{MHA}$ that are distant from multiple negative factors affecting speaker information. The transformed embeddings are then normalized into unit embedding vectors. The ASP generates a single embedding by averaging the normalized embeddings for f using an attention layer that focuses on the important frames.

2.2. Training

2.2.1. Speaker centroid

An important element of the proposed methods is the speaker centroids (prototypes), as described here. To obtain the centroids, we first extract frame-by-frame d-vector embeddings for all utterances in the training data using a pretrained embedder $PE(\cdot)$ and then calculate the centroids $\mathbf{e}^C$ for each speaker i,

$$\mathbf{e}_{i,n}^U = Z\left(\frac{1}{F}\sum_{f=1}^{F} PE\left(\mathbf{x}_{i,n}\right)\right), \quad (5)$$

$$\mathbf{e}_i^C = \frac{1}{N}\sum_{n=1}^{N} \mathbf{e}_{i,n}^U, \quad (6)$$

where n is the utterance index. Utterance-level embeddings $\mathbf{e}_{i,n}^U$ can be obtained by averaging and then normalizing the frame-level embeddings output from a pretrained embedder, and the speaker centroids $\mathbf{e}_i^C$ are derived by averaging these utterance-level embeddings. These calculated speaker centroids serve as references for the training of the embedder.

2.2.2. Loss function

We define loss functions for the embedder and the separator, individually. For the separator, we utilize the scale-invariant signal-to-noise ratio (SI-SNR) loss [20], which is frequently employed in time-domain separators. We use angular prototypical (AP) loss [14] as the embedder loss function. Our training approach applies AP loss with a slight difference compared to the conventional AP loss. In the conventional AP loss, centroids are calculated in every training iteration by averaging the utterance-level embeddings of k samples from the same speaker. However, we pre-calculate these centroids using d-vector embeddings outputs from pretrained embedder and use the fixed centroids as references throughout the training process. The more detailed L_{AP} equation is as follows:

$$\mathcal{L}_{AP} = AP\left(\hat{\mathbf{e}}, \mathbf{e}_{iT}^C\right), \quad (7)$$

where i^T is the target speaker. The AP loss requires careful attention to ensure that the same speaker training sample is not included within a mini-batch, as it performs contrastive learning when there are different speakers within a mini-batch.

To jointly train both the embedder and separator, we apply a weighted sum to the embedding loss $\mathcal{L}_{emb}$ and separator loss $\mathcal{L}_{SI-SNR}$ using the following equation:

$$\mathcal{L} = \omega\mathcal{L}_{AP} + (1-\omega)\,\mathcal{L}_{SI-SNR}, \tag{8}$$

where ω is set to 0.8. The approximate weight was set after several experiments.

2.2.3. Role division training

In TSE task, the embedder and separator fulfill distinct roles. The role of the embedder is to provide accurate target speaker information, whereas the role of the separator is to extract the target speaker with high quality. A large SI-SNR loss indicates the presence of an interference speaker in the output, which, from the perspective mentioned above, is considered an error of the embedder. Furthermore, training with the large SI-SNR loss can hinder convergence. Therefore, we do not train the separator with the large SI-SNR loss. Instead, we train only the embedder in that case. Consequently, we ensure the training stability of the separator by employing a RDT method that excludes training on SI-SNR losses above a certain threshold. This method can be represented as follows:

$$f_{RDT}(x) = \begin{cases} x, & \text{if } x \leq \lambda, \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

where λ represents the threshold for the RDT. It can be fixed or scheduled to vary with training iterations. We configured λ to linearly decrease from 5 to -5 along the training iteration, considering the SIR of the mixture.

When the RDT is applied in joint learning, the loss function can be expressed as follows:

$$\mathcal{L}_{RDT} = \omega\mathcal{L}_{AP} + (1-\omega)\,f_{RDT}(\mathcal{L}_{SI-SNR}), \tag{10}$$

2.2.4. Training strategy

The model is trained using the following strategy:

1. We train the embedder and separator individually. The embedder is trained with the $\mathcal{L}_{AP}$, using $\mathbf{x}^E$ as input, and the separator is trained with $\mathcal{L}_{SI-SNR}$, using $\mathbf{x}^M$ and $\mathbf{e}^C_{i^T}$ as input.
2. The embedder and separator trained in the above step are jointly trained in this step. The proposed model is trained with $\mathcal{L}_{RDT}$, using $\mathbf{x}^M$ and $\mathbf{x}^E$ as input.

3. Experiments

3.1. Datasets

We use LibriSpeech [21], and the training and test datasets are reconstructed using the mixture dataset list of VoiceFilter[1]. This dataset consists of triple sets (enrollment, target, and interference), with approximately 280,000 sets for training and 5,567 sets for testing. In the training dataset, the SIR is set between -5 and 5 dB, and the test set is mixed with the original volume.

[1]https://github.com/google/speaker-id/tree/master/publications/VoiceFilter/dataset/LibriSpeech

The configuration of LibriSpeech DB consists of speaker ID, audio book ID, and utterance index. When the speaker ID and audio book ID are identical, it implies that the environmental elements (such as microphone characteristics, reverberation, background noise) are consistent. Conversely, if the speaker ID is the same, but the audio book ID differs, it indicates that the environments are subtly different. We aim to utilize this aspect to analyze the results based on the match between the enrollment and target utterance environments in the test set. The test set consists of 2,745 cases in which the environments of the enrollment and target are the same, and 2,822 cases where they are different. In the LibriSpeech DB, there are several audio book chapters for each speaker.

3.2. Experimental setup

The MHA has 8 attention heads and 2 layers, and each layer has 256 input and output nodes. The inner layer of the feed-forward network of MHA has 512 nodes. We use a pretrained embedder, which is trained on VoxCeleb2 DB [22], with a publicly available source code and model.[2] Furthermore, we use the Conformer-based ASR model [23], which is trained with GigaSpeech [24] and LibriSpeech.

As shown in Table 1, we test several models to verify the training methods. We set the learning rate to 1e-3 and use cosine decay with a warm-up. In addition, we use the Adam optimizer. For the baseline experiment, the final step is set to 500k steps with 40k warm-up steps. In Step 1 of the training strategy (TS), the model is trained until the loss converges. In Step 2, 350k final steps and 4k warm-up steps are used, applying AP loss up to 250k. The hyper-parameters of the model are applied, as introduced in the previous section. The batch size for the training embedder is 128, the batch size for joint learning is 16.

To evaluate the performance of TSE, we measure scale-invariant signal-to-noise ratio improvement (SI-SNRi) [25], signal-to-distortion ratio improvement (SDRi) [26], perceptual evaluation of speech quality (PESQ) [27], and word error rate (WER), which is measured by ASR module. In addition, we measure the failure rate (negative SDRi rate) for additional performance evaluation analysis.

4. Results

X-TasNet is the baseline, and for convenience, we will refer to the proposed methods, X-TasNet with a Transformer-based speaker embedder and training methods, as X-T-TasNet. Additionally, we will denote as $Diff$ and $Same$ based on whether the environment of the enrollment and target utterance matches or not.

4.1. Baseline result analysis

We conducted experiments with the baseline (X-TasNet) and presented the results on the test set in Figure 2, illustrating them as a scatter plot according to the duration of the enrollment utterance and by the negative SDRi (NSDRi) rate [8]. The NSDRi rate is represented as a cumulative distribution function (CDF), allowing us to observe changes in the NSDRi rate based on the slope. The NSDRi rate results of the $Diff$ is 8.68% and the $Same$ is 4.41%, indicating that the error rate due to differences in the environment of the enrollment and target utterance is approximately twice as high. We discovered that environmental differences introduce negative factor information into

[2]https://github.com/mindslab-ai/voicefilter

Table 1: *Evaluation results of X-TasNet and proposed methods. X-T-TasNet results consist of three experiments based on the lambda value of a role division training (RDT). The NSDRi rate represents all ($Diff$/$Same$) samples. The parameter sizes refer to the embedder and separator, respectively.*

Model	$\mathcal{L}_{SI-SNR}$	$\mathcal{L}_{AP}$	RDT	TS	SDRi (↑)	SI-SNRi (↑)	PESQ (↑)	NSDRi rate (↓)	WER (↓)	Param.
X-TasNet	✓	-	-	-	12.79	12.14	2.44	6.57(8.68/4.41)	12.18	12 / 5
X-T-TasNet	✓	✓	5 → -5	✓	**14.25**	**13.48**	**2.65**	**4.71(6.63/2.73)**	**10.58**	20 / 5
	✓	✓	-5 → 5	✓	14.17	13.39	2.65	4.92(6.87/2.88)	10.68	20 / 5
	✓	✓	5	✓	14.15	13.38	2.64	4.85(6.72/2.88)	10.68	20 / 5
w/o RDT	✓	✓	-	✓	13.90	13.25	2.61	5.35(7.19/3.46)	10.75	20 / 5
w/o RDT, $\mathcal{L}_{AP}$	✓	-	-	✓	13.66	12.97	2.60	6.18(7.55/4.15)	11.25	20 / 5

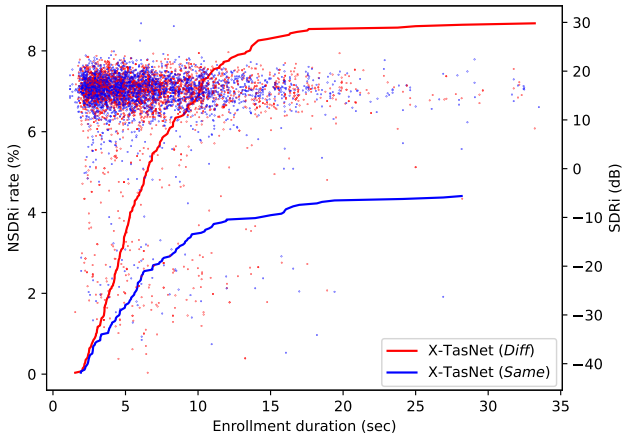

Figure 2: *X-TasNet result. The scatter plot roughly illustrates the distribution of SDRi according to the duration of enrollment utterances in the test sets. The line plots are the CDF of $Diff$, $Same$ NSDRi rates.*

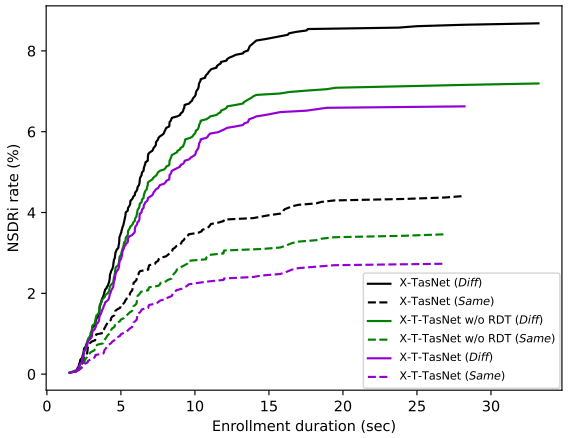

Figure 3: *X-TasNet and proposed methods results. The solid line represents $Diff$, and the dashed line represents $Same$.*

speaker embeddings, thereby degrading the TSE performance. Threfore, we proposed the use of robust speaker embeddings, namely centroids, as a solution.

As you can see in the scatter plot, there are many test cases with short enrollment utterances, resulting in a steep slope of the NSDRi rate. Therefore, the slope of the NSDRi rate is utilized solely for relative performance comparison analysis with other models.

4.2. Comparison with baseline and proposed methods

As shown in Table 1, we conducted an ablation study on our proposed methods and compared them with the baseline across various evaluation metrics. Overall, it is evident that the proposed methods significantly outperform the baseline. Additionally, the performance progressively improves as more of the proposed methods are incorporated. In RDT, reducing the lambda as training progresses leads to better TSE performance than keeping it constant or increasing it.

Figure 3 shows the NSDRi rate for each of the models as a CDF. Comparing the improvement (gap size of each plot) in NSDRi rate for $Diff$ and $Same$ between the X-T-TasNet w/o RDT and X-TasNet, the $Diff$ case shows more improvement than the $Same$ case, due to the centroid estimation by the X-T-TasNet w/o RDT model. Conversely, X-T-TasNet shows more improvement in the $Same$ case. This is because interference speaker outputs are more likely in $Diff$ cases, and RDT tends to exclude $Diff$ cases more in training.

As shown in Figure 3, performance improvement in the $Same$ case starts from very short enrollment, while in the $Diff$ case it begins at about 7 seconds. Although our main goal is to improve the $Diff$ case using centroids, we also observed improvements in the $Same$ case. In the $Same$ case, since the environment is identical, there are no effects from environmental negative factors, but text-dependent information negatively impacts performance. Therefore, the shorter utterance, the more prone it is to text-dependent information bias, and our proposed methods improves this by estimating the centroid.

5. Conclusion

In this paper, we proposed a robust target speaker extraction using a Transformer-based speaker embedder to estimate centroid and a role division training method. Through several experiments, we demonstrated the performance improvement for the proposed methods. In particular, the performance was significantly improved when the environments of the enrollment and target utterances were different. In addition, because only the parameters of the embedder were increased in the proposed architecture, the RTF of the separator for inference is not affected, making it practical and useful.

6. Acknowledgements

This research was supported by Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2024 (Project Name: Development of Co-Pilot technology for automatic completion of generative AI-based 3D Webtoon, Project Number: RS-2024-00400004, Contribution Rate: 30 %)

7. References

[1] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2016, pp. 31–35.

[2] Y. Luo, Z. Chen, and T. Yoshioka, "Dual-path rnn: efficient long sequence modeling for time-domain single-channel speech separation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 46–50.

[3] C. Subakan, M. Ravanelli, S. Cornell, M. Bronzi, and J. Zhong, "Attention is all you need in speech separation," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 21–25.

[4] S. E. Eskimez, T. Yoshioka, H. Wang, X. Wang, Z. Chen, and X. Huang, "Personalized speech enhancement: New models and comprehensive evaluation," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 356–360.

[5] S.-J. Chen, A. S. Subramanian, H. Xu, and S. Watanabe, "Building state-of-the-art distant speech recognition using the chime-4 challenge with a setup of speech enhancement baseline," *Proc. Interspeech 2018*, pp. 1571–1575, 2018.

[6] H. Sato, T. Ochiai, M. Delcroix, K. Kinoshita, T. Moriya, and N. Kamo, "Should we always separate?: Switching between enhanced and observed signals for overlapping speech recognition," *Proc. Interspeech 2021*, pp. 1149—1153, 2021.

[7] Q. Wang, H. Muckenhirn, K. Wilson, P. Sridhar, Z. Wu, J. R. Hershey, R. A. Saurous, R. J. Weiss, Y. Jia, and I. L. Moreno, "Voicefilter: Targeted voice separation by speaker-conditioned spectrogram masking," *Proc. Interspeech 2019*, pp. 2728–2732, 2019.

[8] Z. Zhang, B. He, and Z. Zhang, "X-tasnet: Robust and accurate time-domain speaker extraction network," *Proc. Interspeech 2020*, pp. 1421–1425, 2020.

[9] W. Liu and C. Xie, "Gated convolutional fusion for time-domain target speaker extraction network," *Proc. Interspeech 2022*, pp. 5368–5372, 2022.

[10] M. Delcroix, K. Kinoshita, T. Ochiai, K. Zmolikova, H. Sato, and T. Nakatani, "Listen only to me! how well can target speech extraction handle false alarms?" *Proc. Interspeech 2022*, pp. 1–5, 2022.

[11] K. Liu, Z. Du, X. Wan, and H. Zhou, "X-sepformer: End-to-end speaker extraction network with explicit optimization on speaker confusion," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[12] H. Sato, T. Ochiai, M. Delcroix, K. Kinoshita, T. Moriya, N. Makishima, M. Ihori, T. Tanaka, and R. Masumura, "Strategies to improve robustness of target speech extraction to enrollment variations," *Proc. Interspeech 2022*, pp. 996–1000, 2022.

[13] S. Mun, D. Gowda, J. Lee, C. Han, D. Lee, and C. Kim, "Prototypical speaker-interference loss for target voice separation using non-parallel audio samples," *Proc. Interspeech 2022*, pp. 276–280, 2022.

[14] J. S. Chung, J. Huh, S. Mun, M. Lee, H.-S. Heo, S. Choe, C. Ham, S. Jung, B.-J. Lee, and I. Han, "In defence of metric learning for speaker recognition," *Proc. Interspeech 2020*, pp. 2977–2981, 2020.

[15] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *Proc. Interspeech 2020*, pp. 3830–3834, 2020.

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

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

[18] K. Kishan, Z. Tan, L. Chen, M. Jin, E. Han, A. Stolcke, and C. Lee, "Openfeat: Improving speaker identification by open-set few-shot embedding adaptation with transformer," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7062–7066.

[19] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive statistics pooling for deep speaker embedding," *Proc. Interspeech 2018*, pp. 2252–2256, 2018.

[20] Y. Isik, J. Le Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-channel multi-speaker separation using deep clustering," *Interspeech 2016*, pp. 545–549, 2016.

[21] V. Panayotov, G. Chen, D. Povey, and S. 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.

[22] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *Proc. Interspeech 2018*, pp. 1086–1090, 2018.

[23] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," *Proc. Interspeech 2020*, pp. 5036–5040, 2020.

[24] G. Chen, S. Chai, G. Wang, J. Du, W.-Q. Zhang, C. Weng, D. Su, D. Povey, J. Trmal, J. Zhang *et al.*, "Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio," *Proc. Interspeech 2021*, 2021.

[25] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr–half-baked or well done?" in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 626–630.

[26] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE transactions on audio, speech, and language processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[27] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs," in *2001 IEEE international conference on acoustics, speech, and signal processing (ICASSP)*. IEEE, 2001, pp. 749–752.