人工知能学会第二種研究会資料
AIチャレンジ研究会
SIG-Challenge-066-02

Speech Separation with Auxiliary Signal-to-Artifact Ratio Loss for Improving Multi-Talker ASR

Matthew Ngai[1,2*] Chikara Maeda[1] Muhammad Shakeel[1] Yui Sudo[1]

[1] Honda Research Institute Japan Co., Ltd.
[2] The University of British Columbia

Abstract: Speech separation (SS) is widely used as a front-end to separate individual speech signals from a speech mixture for robust multi-talker automatic speech recognition (ASR). However, the front-end SS processing introduces artifacts in recovered speech signals, leading to the loss of information for off-the-shelf ASR models, resulting in lower ASR performance. In this work, we propose an auxiliary training objective — a weighted-sum of scale-invariant signal-to-artifact ratio (SI-SAR) and signal-to-noise ratio (SI-SNR) loss — which enforces the model to estimate the recovered speech signals with fewer artifacts. We provide an in-depth analysis of its essential behaviors from two perspectives: (1) it encourages the SS model to consider artifact errors in addition to noise and interference errors, thereby improving ASR performance on separated speech; (2) it increases the similarity between the input speech and ASR training data, particularly when a low-power signal-to-noise ratio of Gaussian white noise is added, further enhancing ASR performance. Extensive experiments on the Libri2Mix dataset demonstrate the effectiveness of our auxiliary loss. Notably, when using off-the-shelf ASR models, the proposed method achieves an absolute word error rate (WER) of 4.46% and 12.36% on the `mix clean` and `mix both` test sets, respectively. Furthermore, adding Gaussian white noise with a signal-to-noise ratio of 24dB to estimated speech signals results in an absolute improvement in WER by 0.2% – 0.4%.

1 Introduction

Current research in speech separation (SS) [1–5] has demonstrated impressive separation capabilities and has served as a front-end in many modular speech processing systems [6–8], from automatic meeting transcription to conversational AI. These systems are often modular and can thus leverage advancements within their individual modules, thanks to the open-source availability [9–12] of state-of-the-art separation and speech foundation models (SFM) [13–18]. While modular systems offer flexibility by processing the speech in a cascaded manner, they suffer from information loss from each independent module, resulting in a low automatic speech recognition (ASR) performance. This ASR degradation is significant when front-end SS models are deployed in diverse acoustic environments and introduce unnatural distortions and artifacts in recovered speech [19–21]. To maintain high performance in various acoustic environments, practitioners must perform independent (re-)training of individual SS or ASR modules, leading to higher computational costs.

To improve the modular speech processing system performance, in this work, we propose the weighted-sum of scale-invariant signal-to-artifact ratio (SI-SAR) and signal-to-noise ratio (SI-SNR) loss, employed on the SS model as a regularized objective. It estimates the recovered speech signals with fewer artifacts and minimizes the information loss between the front-end and ASR modules. We analyze its effect from the following two perspectives. First, it encourages the SS model to consider artifact errors in addition to noise and interference errors, thereby improving recovered speech intelligibility for ASR modules. Second, we artificially include Gaussian white noise with specific signal-to-noise ratios (SNR) to mask artifacts introduced by SS and increase the similarity between the separated signals and ASR training data commonly used to represent noisy real-world conditions.

*Contact: Honda Research Institute Japan Co., Ltd.
8-1 Honcho, Wako-shi, Saitama 351-0188, Japan
E-mail: matthew.ngai@jp.honda-ri.com

2 Related work

Several studies have addressed the challenge of incorporating SS models in modular speech processing systems as a front-end to mitigate the effects of information loss prior to ASR. These efforts can be broadly categorized into two approaches:

(1) Joint (re-)training-based methods [22,23] which involve jointly optimizing the SS and ASR objectives to ensure the separated speech is better aligned with the off-the-shelf SFM model [13–18]. This strategy effectively improves recognition performance. However, it is not feasible to (re-)train the SFM model due to the high training cost and the risk of performance degradation on out-of-domain datasets when deployed in different application scenarios or acoustic environments. In addition, joint (re-)training can reduce the ASR module's ability to handle distortions [24].

(2) Retraining-free methods directly optimize SS models, often employing auxiliary losses [19] to mitigate the effects of distortion and artifacts in the recovered speech. A similar approach, involving an auxiliary "artifact-boosted" training loss [21], has been found beneficial for SS models and has shown improved ASR performance. However, in their multi-talker scenario, the interfering speaker's speech was added to the mixture with lower signal power, which does not reflect true SS scenarios. Other methods mitigate the ASR performance degradation due to artifacts by using Monte Carlo-based feature estimation [25, 26]. However, they require multiple Monte Carlo sampling processes, resulting in high computational costs.

3 Proposed Method

We first define speech mixtures and the metrics used to evaluate SS and ASR performance in Sections 3.1-3.3. Then, we introduce our proposed SI-SAR auxiliary loss, and noise adding in Sections 3.4-3.5.

3.1 Speech Mixtures

A time-domain mixture with J speakers can be modeled as a vector $\mathbf{x} \in \mathbb{R}^T$ for the mixture, vectors $\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_J \in \mathbb{R}^T$ for the sources, and $\mathbf{n} \in \mathbb{R}^T$ for noise:

$$\mathbf{x} = \sum_{j=1}^{J} \mathbf{s}_j + \mathbf{n}, \tag{1}$$

where T is the number of samples in a given signal.

3.2 Speech Separation

The single-channel SS task aims to separate each speaker into source signals $\widehat{\mathbf{s}}_1, \widehat{\mathbf{s}}_2, ..., \widehat{\mathbf{s}}_J \in \mathbb{R}^T$. In this study, the noise in the original mixture, if any, is not an output of SS.

To quantify the performance of the separated source estimates, we use the blind source separation evaluation metrics proposed by Vincent et al. [27]. In [27], the estimated source signal for speaker index j, $\widehat{\mathbf{s}}_j$, is described as:

$$\widehat{\mathbf{s}}_j = \mathbf{s}_{\text{target}} + \mathbf{e}_{\text{interf}} + \mathbf{e}_{\text{noise}} + \mathbf{e}_{\text{artif}}, \tag{2}$$

where $\mathbf{s}_{\text{target}}$ is the clean speech signal of speaker j. Qualitatively, the error terms describe: $\mathbf{e}_{\text{interf}}$, the residual speech of speakers that are not the target; $\mathbf{e}_{\text{noise}}$, the residual noise that was present in the original mixture; and $\mathbf{e}_{\text{artif}}$, any other errors that were not present in the original mixture.

To compute the error terms, let $\mathbf{P}_{\mathbf{s}_j}, \mathbf{P}_{\mathbf{s}}, \mathbf{P}_{\mathbf{s},\mathbf{n}}$ be orthogonal projection matrices. $\mathbf{P}_{\mathbf{s}_j}$ projects onto the subspaces of a given speaker, $\mathbf{P}_{\mathbf{s}}$ projects onto the subspace of all speakers present in the mixture, and $\mathbf{P}_{\mathbf{s},\mathbf{n}}$ projects onto the subspace of all speakers present including noise. Then, [27] defines the target source signal and error terms as:

$$\mathbf{s}_{\text{target}} \coloneqq \mathbf{P}_{\mathbf{s}_j} \widehat{\mathbf{s}}_j, \tag{3}$$

$$\mathbf{e}_{\text{interf}} \coloneqq \mathbf{P}_{\mathbf{s}} \widehat{\mathbf{s}}_j - \mathbf{P}_{\mathbf{s}_j} \widehat{\mathbf{s}}_j, \tag{4}$$

$$\mathbf{e}_{\text{noise}} \coloneqq \mathbf{P}_{\mathbf{s},\mathbf{n}} \widehat{\mathbf{s}}_j - \mathbf{P}_{\mathbf{s}} \widehat{\mathbf{s}}_j, \tag{5}$$

$$\mathbf{e}_{\text{artif}} \coloneqq \widehat{\mathbf{s}}_j - \mathbf{P}_{\mathbf{s},\mathbf{n}} \widehat{\mathbf{s}}_j. \tag{6}$$

Following those definitions, the metrics source-to-noise ratio (SNR), source-to-distortion ratio (SDR), sources-to-interferences ratio (SIR), and sources-to-artifacts ratio are defined as:

$$\text{SNR} \coloneqq 10 \log_{10} \frac{\left\| \mathbf{s}_{\text{target}} + \mathbf{e}_{\text{interf}} \right\|^2}{\left\| \mathbf{e}_{\text{noise}} \right\|^2}, \tag{7}$$

$$\text{SDR} \coloneqq 10 \log_{10} \frac{\left\| \mathbf{s}_{\text{target}} \right\|^2}{\left\| \mathbf{e}_{\text{interf}} + \mathbf{e}_{\text{noise}} + \mathbf{e}_{\text{artif}} \right\|^2}, \tag{8}$$

$$\text{SIR} \coloneqq 10 \log_{10} \frac{\left\| \mathbf{s}_{\text{target}} \right\|^2}{\left\| \mathbf{e}_{\text{interf}} \right\|^2}, \tag{9}$$

$$\text{SAR} \coloneqq 10 \log_{10} \frac{\left\| \mathbf{s}_{\text{target}} + \mathbf{e}_{\text{interf}} + \mathbf{e}_{\text{noise}} \right\|^2}{\left\| \mathbf{e}_{\text{artif}} \right\|^2}. \tag{10}$$

Further, the scale-invariant versions of these metrics (SI-SNR, SI-SDR, SI-SIR, SI-SAR) proposed by [28] yield the same ratio regardless of the vector magnitudes of the target and estimated speech. The SI-SNR is commonly used as the loss function for training many SS models, including the SepFormer, which is provided in the SpeechBrain toolkit [2, 3, 9, 10].

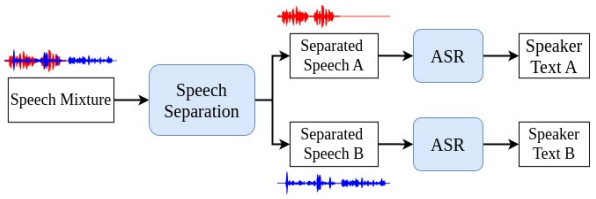

図 1: Cascading separation to recognition pipeline.

3.3 Automatic Speech Recognition

The ASR model converts input speech signals into text sequences. ASR performance is typically evaluated using word error rate (WER) based on the Levenshtein distance [29]. In multi-talker scenarios with overlapping speech, SS models are often used as a front-end to separate overlapped speech signals before feeding them into the ASR model. Note that the front-end SS models often introduce unnatural distortions and artifacts, which degrade ASR performance.

3.4 SI-SAR Auxiliary Loss

To reduce artifact errors, we propose loss function shown in Eq. (11) with permutation invariant training (PIT) [30] which incorporates the SI-SAR with weight λ along with the SI-SNR.

$$\begin{aligned}\mathcal{L} = [-\lambda \times \text{SI-SAR}(\text{ref}, \text{est})] + \\ [(\lambda - 1) \times \text{SI-SNR}(\text{ref}, \text{est})].\end{aligned} \tag{11}$$

In our experiments, we will empirically determine if WER improves for weights $\lambda \in \{0, 0.2, 0.4, 0.6, 0.8\}$.

3.5 Noise Adding

To further mitigate the impact of artifacts, we also investigate adding a small signal power of white noise to the separated speech estimates. This is inspired by a simple technique to mitigate artifacts in the Speech Enhancement (SE) task which is Observation Adding (OA) [19], where a small percentage of the original noisy signal is added to the enhanced signal. For the SS task, OA proves to be ineffective as the ASR model recognizes the speakers that have been added back in after being removed from the original mixture. Despite this, adding a random Gaussian distributed white noise may improve ASR performance by "filling-in" the artifacts caused by the separation model, ultimately improving ASR performance.

4 Experimental Setup

As shown in Figure 1, our separation-recognition pipeline uses independently trained SS and ASR models. This method does not require fine-tuning the entire pipeline, allowing for faster training, flexibility, and modularity for existing ASR systems.

4.1 Dataset

In this study, we used the open-source LibriMix corpus [31] for training and evaluating the separation model. The dataset contains two and three-speaker mixtures derived from LibriSpeech [32] utterances, so we obtain the transcripts for WER evaluation from LibriSpeech. For both two and three speaker mixtures, LibriMix has: two disjoint training sets (`train-100`, `train-360`), one validation set (`dev`), and one test set (`test`). In this paper, we create `train-460` which is the union of the `train-100` and `train-360` datasets. Each dataset contains the following mixing modes `mix_clean` mixtures with overlapping utterances and no noise; `mix_both`, mixtures with overlapping utterances and noise; and `mix_single`, mixtures with a single utterance and noise.

4.2 Separation Model

For the separation task, we used the SepFormer [2, 3] implementation offered in the SpeechBrain toolkit [9, 10] as it is available open-source and allows for reproducibility. Similar to most separation models, the model architecture used in [2] consists of an encoder, masking net, and decoder.

The encoder is a convolutional layer that outputs a 2D STFT-like representation of the mixture signal. The masking net estimates a mask for each speaker to be separated from the STFT-like representation. Finally, the decoder takes the masked 2D representations of each speaker and yields the separated audio signals.

To reduce training time, we trained a set of five SepFormer's on the smaller Libri2Mix `train-100` set in min mode at 16kHz to determine the optimal λ.

Once the optimal λ is determined, we will use it to compare the WER of separated speech from the default 100% SI-SNR SepFormer versus the proposed training objective. However, this time the SepFormer will be trained with the larger Libri2Mix `train-460` set in max mode at 16kHz with noise from WHAM! dataset [33].

表 1: Weighted SI-SNR and SI-SAR combination training results. Training data is specified above each set of models. Evaluation was done with Libri2Mix `wav16k/max/test/mix_clean` set, which contains 3,000 two-speaker mixtures with transcripts from LibriSpeech. Source separation metrics (SI-SNR, SI-SDR, SI-SIR, SI-SAR) are in decibels (dB), higher is better. ASR metric (WER) is in percent, lower is better. These models were trained for 100 epochs, with the same random seed, on A100 80GB GPUs for the smaller `train-100` dataset, and H100 80GB GPUs for the larger `train-460` dataset

System	Training Loss	SI-SNR↑	SI-SDR↑	SI-SIR↑	SI-SAR↑	WER↓	
						RNN-T	Whisper
Training Data: `Libri2Mix/wav16k/min/train-100`							
A	SI-SNR	18.15	18.16	52.67	18.22	8.35	7.85
B	0.2×SI-SAR+0.8×SI-SNR	18.33	18.25	53.13	18.30	7.38	7.32
C	0.4×SI-SAR+0.6×SI-SNR	17.66	17.59	51.63	17.65	9.12	7.94
D	0.6×SI-SAR+0.4×SI-SNR	17.84	17.77	52.00	17.83	8.69	7.56
E	0.8×SI-SAR+0.2×SI-SNR	18.23	18.15	52.64	18.19	7.64	7.22
Training Data: `Libri2Mix/wav16k/max/train-460`							
F	SI-SNR	20.43	20.43	58.15	20.44	4.49	5.35
G	0.2×SI-SAR+0.8×SI-SNR	20.11	20.10	57.51	20.11	**4.46**	**5.30**

表 2: Weighted SI-SNR and SI-SAR combination training results, similar to Table 1. Instead, models in this table were trained with `Libri2Mix/wav16k/max/train-460` including noise from WHAM!. Evaluation was performed with Libri2Mix `wav16k/max/test/mix_both` and transcripts from LibriSpeech. Models were trained for 100 epochs, with the same seed, on H100 80GB GPUs.

System	Training Loss	SI-SNR↑	SI-SDR↑	SI-SIR↑	SI-SAR↑	WER↓	
						RNN-T	Whisper
H	SI-SNR	13.29	13.29	54.83	13.30	12.48	**10.40**
I	0.2×SI-SAR+0.8×SI-SNR	13.45	13.45	55.03	13.46	**12.36**	10.43

The SepFormer model was trained using the loss function defined in (11) with PIT. These models were trained for 100 epochs, with the same random seed, on NVIDIA A100 80GB GPUs for the smaller `train-100` dataset, and NVIDIA H100 80GB GPUs for the larger `train-460` dataset. We used the `fast_bss_eval`[1] [34] toolkit to implement the loss function and evaluate the separation results.

4.3 ASR Model

For the ASR task, we used a pre-trained Recurrent Neural Network Transducer (RNN-T) [35] model from ESPNet trained on 960 hours of utterances from the LibriSpeech corpus. The baseline performance of this ASR model is 3.05% WER on the test-clean set.

Further, to confirm that our proposed method improves the separation-recognition performance for any pre-trained ASR model, we also evaluated WER performance on OpenAI's Whisper-Large-v3 ASR model [13].

[1] `https://github.com/fakufaku/fast_bss_eval`

To solve the permutation problem of not knowing which separated speech estimate corresponds to each speaker transcription, we computed the WER of all combinations of separated speech estimates and speaker transcriptions, then selected the lowest WER.

5 Experimental Results

5.1 SI-SAR Auxiliary Loss Results

As shown in Table 1, adding a percentage of SI-SAR in the training objective can improve ASR performance. In the cases where the ASR performance is degraded (Systems C and D) we believe that the weights off SI-SNR and SI-SAR were too equally divided, so neither system could be fully optimized in the same number of training epochs.

In our results with the RNN-T ASR model, we see a relative improvement of 11.62% WER when we use 20% of the SI-SAR auxiliary loss with the small `train-100` dataset. In the models trained on the

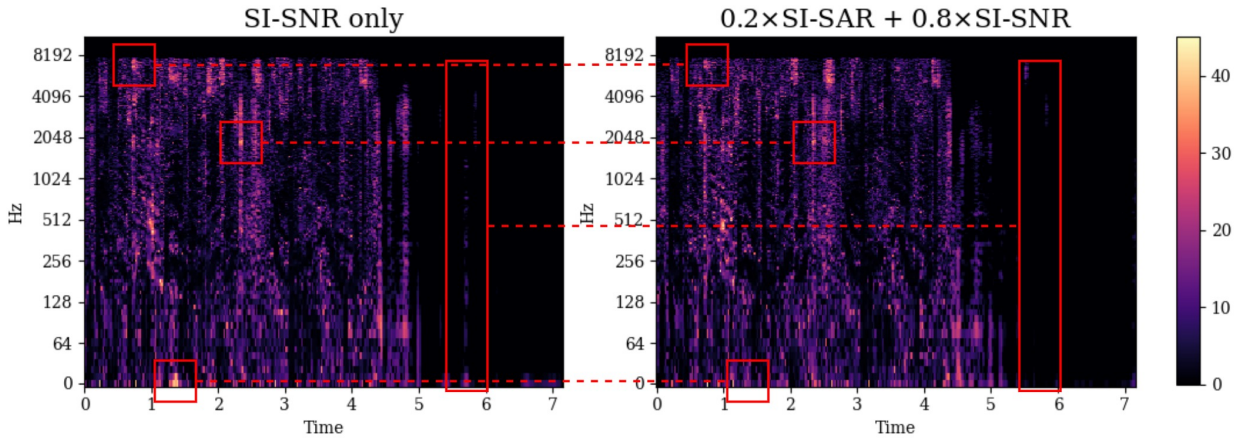

図 2: Spectrograms of the residual error obtained from the absolute difference of the ground truth and estimated separated speech spectrograms. The magnitude is shown in decibels (dB), lower is better. The left image shows the spectrogram residual of separated speech from a SepFormer trained with only SI-SNR loss (System F). The right image shows the residual with 20% weighted SI-SAR loss (System G). The red outlines highlight the artifacts that are more apparent when there is no SI-SAR loss, leading to degraded WER performance.

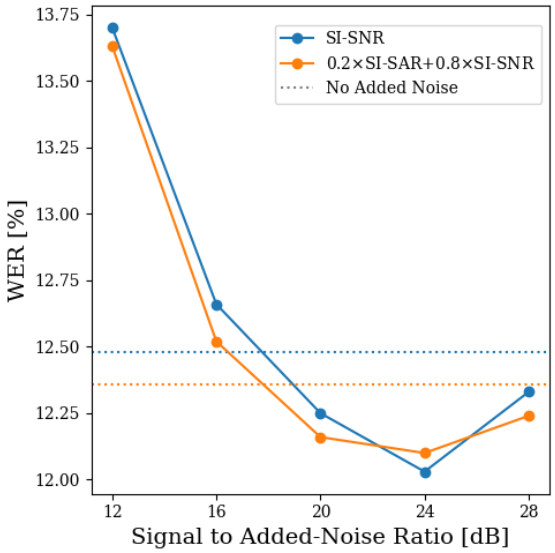

図 3: WER performance with varying signal-to-noise power ratios of additive Gaussian white noise. The horizontal axis is the signal-to-noise ratio, in dB, with respect to the signal power of the separated speaker estimate power to the Gaussian white noise.

larger `train-460` dataset, we see a relative improvement of 0.67% for the models trained without noise (Table 1), and 0.96% relative improvement for the models trained with noise in (Table 2).

We see a similar trend of WER improvement with the 20% SI-SAR auxiliary loss when using the Whisper ASR model except in the case of models trained with `train-460` with noise. However, because Whisper is not trained strictly for the LibriSpeech corpus, it transcribes audio into words that are not identical to the ground truth LibriSpeech transcriptions, even though the output is semantically equivalent. For example, the words "`9000`" and "`NINE THOUSAND`" are different ways of transcribing the same spoken words. In this case, Whisper outputs the former, whereas the LibriSpeech ground truth expects the latter, and so the WER is artificially degraded. At the moment, this discrepancy is acceptable since we still observe a WER improvement on average when the 20% SI-SAR auxiliary loss is applied in separation training.

Figure 2 illustrates the effect of the SI-SAR auxiliary loss. By subtracting the ground truth spectrogram from the separated speech estimate, we observe that the absolute value of the remaining artifacts are more apparent when the separation model is trained on SI-SNR only. In this utterance, using the SI-SAR auxiliary loss yielded an absolute WER improvement from 31.25% (left) to 18.75% (right).

5.2 Noise Adding Results

Figure 3 shows the effect of noise adding on ASR performance. The orange and blue lines represent the results with and without the proposed SI-SAR loss, respectively. The dotted horizontal line indicates the WER of the separated speech estimate without any added noise. We see that when we add Gaussian white noise with certain signal-to-noise ratios, it leads to

even lower WER; we observed an absolute improvement of 0.26% WER (orange) and 0.45% WER (blue). The best WER is achieved when the signal-to-noise ratio is 24dB.

This can be explained by the noise increasing the similarity between the input audio signal and the training set of ASR models. That is, the noise "fills in" or masks some artifacts so the ASR model can recognize the separated speech better, especially if the ASR model is trained on signals with noise.

Notably, the overall improvement achieved through the introduction of the proposed SI-SAR loss is maintained even with noise adding, demonstrating the robustness of our proposed approach. Although the WER at 24 dB slightly favors the model without SI-SAR loss, the benefits of SI-SAR loss remain evident across other conditions.

表 3: Comparison of results from recent Multi-Talker ASR studies and our proposed method. All values are in % WER, lower is better. The sets `test-clean` and `test-both` refer to the `mix_clean` and `mix_both` mixture modes in the test set, respectively.

	Libri2Mix	
	`test-clean`	`test-both`
Fazel and Hsu [36]	7.8	–
Polok et al. [37]	–	17.6
Meng et al. [38]	4.66	–
System G (Ours)	**4.46**	30.82
System I (Ours)	4.67	**12.36**

We also compare our results with recent works involving Multi-Talker ASR using various architectures on the Libri2Mix dataset in Table 3. Not only is it clear that the SI-SAR auxiliary training loss successfully mitigates artifact errors, but our results also show that SS front-ends in cascaded systems can achieve competitive Multi-Talker ASR performance.

6 Conclusion

In this paper, we investigated the use of an auxiliary loss, SI-SAR, for training a Transformer-based speech separation model with the goal of improving WER performance. Our experimental results showed that a weighted sum of 80% SI-SNR and 20% SI-SAR training objective achieves optimal WER performance with a relative WER improvement of 11.62% with separation models trained on small datasets, and up to 0.96% improvement with models trained on larger datasets.

We also found that adding a low amount (24dB SNR) of Gaussian white noise before the ASR step is a simple way to improve the absolute WER by about 0.2% – 0.4% while maintaining improvements with our proposed auxiliary loss.

In future studies, the effect of SI-SAR auxiliary loss can be confirmed by applying it to other Transformer-based separation models such as MossFormer [5, 39]. Also, evaluating with more ASR models can further support our findings, given that the output of the ASR models are compatible with the ground truth transcriptions.

参考文献

[1] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[2] C. Subakan, M. Ravanelli, S. Cornell, M. Bronzi, and J. Zhong, "Attention is all you need in speech separation," in *Proc. ICASSP*. IEEE, 2021, pp. 21–25.

[3] C. Subakan, M. Ravanelli, S. Cornell, F. Grondin, and M. Bronzi, "Exploring self-attention mechanisms for speech separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 2169–2180, 2023.

[4] Z.-Q. Wang, S. Cornell, S. Choi, Y. Lee, B.-Y. Kim, and S. Watanabe, "TF-GRIDNET: Making time-frequency domain models great again for monaural speaker separation," in *Proc. ICASSP*. IEEE, 2023, pp. 1–5.

[5] S. Zhao, Y. Ma, C. Ni, C. Zhang, H. Wang, T. H. Nguyen, K. Zhou, J. Q. Yip, D. Ng, and B. Ma, "Mossformer2: Combining transformer and rnn-free recurrent network for enhanced time-domain monaural speech separation," in *Proc. ICASSP*. IEEE, 2024, pp. 10 356–10 360.

[6] S. Watanabe, M. Mandel, J. Barker, E. Vincent, A. Arora, X. Chang, S. Khudanpur, V. Manohar, D. Povey, D. Raj, D. Snyder, A. S. Subramanian, J. Trmal, B. B. Yair, C. Boeddeker, Z. Ni, Y. Fujita, S. Horiguchi, N. Kanda, T. Yoshioka, and N. Ryant, "Chime-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings," in *Proc. Interspeech*, 2020, pp. 1–7.

[7] Z. Chen, T. Yoshioka, L. Lu, T. Zhou, Z. Meng, Y. Luo, J. Wu, X. Xiao, and J. Li, "Continuous speech separation: Dataset and analysis," in *Proc. ICASSP*. IEEE, 2020, pp. 7284–7288.

[8] D. Raj, P. Denisov, Z. Chen, H. Erdogan, Z. Huang, M. He, S. Watanabe, J. Du, T. Yoshioka, Y. Luo, N. Kanda, J. Li, S. Wisdom, and J. R. Hershey, "Integration of speech separation, diarization, and recognition for multi-speaker meetings: System description, comparison, and analysis," in *Proc. SLT*. IEEE, 2021, pp. 897–904.

[9] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, L. Lugosch, C. Subakan, N. Dawalatabad, A. Heba, J. Zhong *et al.*, "Speechbrain: A general-purpose speech toolkit," *arXiv preprint arXiv:2106.04624*, 2021.

[10] M. Ravanelli, T. Parcollet, A. Moumen, S. de Langen, C. Subakan, P. Plantinga, Y. Wang, P. Mousavi, L. Della Libera, A. Ploujnikov *et al.*, "Open-source conversational AI with speechbrain 1.0," *Journal of Machine Learning Research*, vol. 25, no. 333, pp. 1–11, 2024.

[11] M. Pariente, S. Cornell, J. Cosentino, S. Sivasankaran, E. Tzinis, J. Heitkaemper, M. Olvera, F.-R. Stöter, M. Hu, J. M. Martín-Doñas, D. Ditter, A. Frank, A. Deleforge, and E. Vincent, "Asteroid: the PyTorch-based audio source separation toolkit for researchers," in *Proc. Interspeech*, 2020.

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

[13] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *International conference on machine learning*. PMLR, 2023, pp. 28 492–28 518.

[14] Y. Peng, J. Tian, B. Yan, D. Berrebbi, X. Chang, X. Li, J. Shi, S. Arora, W. Chen, R. Sharma *et al.*, "Reproducing whisper-style training using an open-source toolkit and publicly available data," in *Proc. ASRU*, 2023, pp. 1–8.

[15] Y. Peng, J. Tian, W. Chen, S. Arora, B. Yan, Y. Sudo, S. Muhammad, K. Choi, J. Shi, X. Chang *et al.*, "OWSM v3.1: Better and faster open whisper-style speech models based on E-Branchformer," in *Proc. Interspeech*, 2024, pp. 352–356.

[16] Y. Peng, Y. Sudo, M. Shakeel, and S. Watanabe, "OWSM-CTC: An open encoder-only speech foundation model for speech recognition, translation, and language identification," in *Proc. ACL*, 2024, pp. 10 192–10 209.

[17] B. Zhang, H. Lv, P. Guo, Q. Shao, C. Yang, L. Xie, X. Xu, H. Bu, X. Chen, C. Zeng *et al.*, "Wenetspeech: A 10000+ hours multi-domain mandarin corpus for speech recognition," in *Proc. ICASSP*, 2022, pp. 6182–6186.

[18] Y. Yin, D. Mori, and S. Fujimoto, "Reazonspeech: A free and massive corpus for Japanese ASR," in *The Association for Natural Language Processing*, 2023, pp. 1134–1139.

[19] K. Iwamoto, T. Ochiai, M. Delcroix, R. Ikeshita, H. Sato, S. Araki, and S. Katagiri, "How bad are artifacts?: Analyzing the impact of speech enhancement errors on ASR," in *Proc. Interspeech*, 2022, pp. 5418–5422.

[20] ——, "How does end-to-end speech recognition training impact speech enhancement artifacts?" in *Proc. ICASSP*, 2024, pp. 11 031–11 035.

[21] T. Ochiai, K. Iwamoto, M. Delcroix, R. Ikeshita, H. Sato, S. Araki, and S. Katagiri, "Rethinking processing distortions: Disentangling the impact of speech enhancement errors on speech recognition performance," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 3589–3602, 2024.

[22] T. von Neumann, K. Kinoshita, L. Drude, C. Boeddeker, M. Delcroix, T. Nakatani, and R. Haeb-Umbach, "End-to-end training of time domain audio separation and recognition," in *Proc. ICASSP*, 2020, pp. 7004–7008.

[23] T. von Neumann, C. Boeddeker, L. Drude, K. Kinoshita, M. Delcroix, T. Nakatani, and R. Haeb-Umbach, "Multi-talker ASR for an unknown number of sources: Joint training of source counting, separation and asr," in *Proc. Interspeech*, 2020, pp. 3097–3101.

[24] P. Wang, K. Tan, and D. Wang, "Bridging the gap between monaural speech enhancement and recognition with distortion-independent acoustic modeling," in *Proc. Interspeech*, 2019, pp. 471–475.

[25] R. Takeda, Y. Sudo, K. Nakadai, and K. Komatani, "Empirical sampling from latent utterance-wise evidence model for missing data ASR based on neural encoder-decoder model," in *Proc. Interspeech*, 2022, pp. 3789–3793.

[26] R. Takeda, Y. Sudo, and K. Komatani, "Flexible evidence model to reduce uncertainty mismatch between speech enhancement and ASR based on encoder-decoder architecture," in *Proc. APSIPA ASC*, 2023.

[27] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[28] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "SDR–half-baked or well done?" in *Proc. ICASSP*. IEEE, 2019, pp. 626–630.

[29] L. Yujian and L. Bo, "A normalized levenshtein distance metric," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1091–1095, 2007.

[30] M. Kolbæk, D. Yu, Z.-H. Tan, and J. Jensen, "Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 10, pp. 1901–1913, 2017.

[31] J. Cosentino, M. Pariente, S. Cornell, A. Deleforge, and E. Vincent, "LibriMix: An open-source dataset for generalizable speech separation," 2020.

[32] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *Proc. ICASSP*, 2015.

[33] G. Wichern, E. McQuinn, J. Antognini, M. Flynn, R. Zhu, D. Crow, E. Manilow, and J. Le Roux, "Wham!: Extending speech separation to noisy environments," in *Proc. Interspeech*, 2019, pp. 1368–1372.

[34] R. Scheibler, "SDR—Medium rare with fast computations," in *Proc. ICASSP*. IEEE, 2022, pp. 701–705.

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

[36] M. Fazel-Zarandi and W.-N. Hsu, "Cocktail hubert: Generalized self-supervised pre-training for mixture and single-source speech," in *Proc. ICASSP*. IEEE, 2023, pp. 1–5.

[37] A. Polok, D. Klement, M. Wiesner, S. Khudanpur, J. Černocký, and L. Burget, "Target speaker ASR with Whisper," 2024. [Online]. Available: https://arxiv.org/abs/2409.09543

[38] L. Meng, J. Kang, Y. Wang, Z. Jin, X. Wu, X. Liu, and H. Meng, "Empowering whisper as a joint multi-talker and target-talker speech recognition system," in *Proc. Interspeech*, 2024, pp. 4653–4657.

[39] S. Zhao and B. Ma, "Mossformer: Pushing the performance limit of monaural speech separation using gated single-head transformer with convolution-augmented joint self-attentions," in *Proc. ICASSP*. IEEE, 2023, pp. 1–5.