Unified Modeling of Multi-Talker Overlapped Speech Recognition and Diarization with a Sidecar Separator

Lingwei Meng[1], Jiawen Kang[1], Mingyu Cui[1], Haibin Wu[2], Xixin Wu[1], Helen Meng[1]

[1] Dept. of Systems Engineering & Engineering Management, The Chinese University of Hong Kong
[2] Graduate Institute of Communication Engineering, National Taiwan University

{lmeng, jwkang, mycui, wuxx, hmmeng}@se.cuhk.edu.hk, f07921092@ntu.edu.tw

Abstract

Multi-talker overlapped speech poses a significant challenge for speech recognition and diarization. Recent research indicated that these two tasks are inter-dependent and complementary, motivating us to explore a unified modeling method to address them in the context of overlapped speech.

A recent study proposed a cost-effective method to convert a single-talker automatic speech recognition (ASR) system into a multi-talker one, by inserting a *Sidecar* separator into the frozen well-trained ASR model. Extending on this, we incorporate a diarization branch into the Sidecar, allowing for unified modeling of both ASR and diarization with a negligible overhead of only 768 parameters. The proposed method yields better ASR results compared to the baseline on LibriMix and LibriSpeechMix datasets. Moreover, without sophisticated customization on the diarization task, our method achieves acceptable diarization results on the two-speaker subset of CALL-HOME with only a few adaptation steps.

Index Terms: multi-talker speech recognition, end-to-end speech recognition, domain adaptation, speaker diarization

1. Introduction

Multi-talker (or multi-speaker) overlapped speech has presented significant challenges for many tasks in speech techniques, such as automatic speech recognition (ASR) and diarization [1]. Although recent years have seen progress in addressing this scenario, these studies tend to be conducted independently, isolated within their respective fields.

In the field of multi-talker overlapped speech recognition, two dominant paradigms have emerged: cascade architectures and fully end-to-end models, both with their drawbacks [1]. The cascade architectures, where an ASR module follows a speech separation module, require joint training [2, 3], and may cause performance degradation in the modules' original domains. In contrast, the carefully customized end-to-end models typically necessitate extensive training efforts from scratch [4, 5, 6, 7], and fail to capitalize on significant achievements made in common single-talker ASR. To overcome the limitations of current multi-talker ASR paradigms, a novel **Sidecar** approach has been recently proposed [8]. It involves loosely coupling with well-trained single-talker ASR models, which are then efficiently adapted for multi-talker scenes, without altering the original model's parameters. Specifically, the Sidecar approach converts a single-talker wav2vec 2.0-based ASR model into a multi-talker ASR model, by plugging a Sidecar separator between two lower encoder layers. This integration utilizes speech separation techniques to effectively disentangle overlapped acoustic embeddings from multiple speakers, thus equipping a standard ASR system to manage multi-talker ASR

at a minimal cost. This research inspires us that the representations hierarchically extracted by ASR encoder [9, 10] can be leveraged to handle multiple tasks in a cost-effective manner.

In the diarization field, end-to-end modeling methods have emerged as a promising alternative in recent years [11, 12, 13, 14], demonstrating superiority in handling overlapped speech compared to conventional pipelines based on speaker embedding clustering [15, 16, 17]. Although limited in number, some recent studies have investigated joint multi-talker speech recognition and diarization modeling, and indicated that the two tasks are inter-dependent and complementary [18]. Methods such as [19], [20], and speaker-attributed ASR [21, 22] predict ASR transcriptions alongside sentence-level speaker identification. However, these works do not explicitly output timestamps for speaker activity boundaries. [23] proposes to iteratively apply an external speaker embedding extractor and a target-speaker ASR model; and similarly the pipeline in [24] requires an external pre-trained speaker embedding extractor. They showed promising results in ASR with additional timestamps, but their systems are intricate and not truly unified in modeling the ASR and diarization tasks. RPN-JOINT [25] is a cascade architecture consisting of a diarization module followed by an ASR module with shared lower blocks. The modules are pre-trained in their respective domains and subsequently fine-tuned jointly. However, since the majority of the two modules remain separate, the overall architecture can still be quite cumbersome.

Encouraged by the capability of the Sidecar in separating embeddings, we aim to extend the prospects of its application for low-cost, end-to-end unified modeling of ASR and diarization. Instead of deploying individual ASR and diarization modules, we hypothesize that a unified backbone could foster knowledge sharing between the two tasks. As shown in Figure 1, building upon the Sidecar approach, we incorporate a diarization branch with merely 768 additional parameters, thereby enabling the unified modeling of both ASR and diarization tasks with negligible computational overhead. The total number of trainable parameters is 8.7 M (8.4% of all parameters) for the two-speaker model and 8.8 M (8.5% of all parameters) for the three-speaker model. The contributions of the proposed method are threefold:

- We propose a pioneering framework for unified modeling multi-talker ASR and diarization tasks. Exploiting the frozen well-trained ASR model, this approach only contains a small number of trainable parameters hence easy to implement.

- Leveraging the inter-dependence between ASR and diarization tasks, our proposed method outperforms the original Sidecar scheme for both two- and three-speaker overlapped speech recognition tasks. We demonstrate that this strategy is not only feasible with wav2vec 2.0 backbone, but also with data2vec 2.0 [26, 27] - yielding even better results for ASR.

- Furthermore, without any sophisticated customization on diarization task, our proposed method achieves acceptable performance on the two-speaker subset of CALLHOME with only a few adaptation steps.

We believe that the proposed method holds the potential for a cost-effective solution for the unified modeling of multi-talker overlapped speech recognition and diarization.

2. Unified Modeling of Multi-Talker Speech Recognition and Diarization with Sidecar

The proposed approach comprises three main components: a well-trained single-talker ASR model with the parameters frozen, a Sidecar separator with diarization branch, and the training objective. Figure 1 illustrates that the Sidecar with diarization branch is inserted between two ASR encoder layers, aided by one convolutional layer on each side, creating a multi-talker ASR and diarization unified modeling system. The model is optimized with permutation invariant training (PIT) [28] for connectionist temporal classification (CTC) loss [29], and using the same permutation for the diarization loss.

No lexicons or language models are involved in this work.

2.1. Well-trained single-talker ASR model

An end-to-end ASR model typically comprises an encoder that converts waveform or acoustic features into high-level representations and a decoder that models these representations into language tokens. However, training such a model from scratch can be time-consuming and challenging, especially in multi-talker environments. As indicated in [8], a Sidecar separator can re-purpose existing single-talker models for multi-talker overlapped speech recognition with low cost.

As a well-known pre-trained speech representation model based on self-supervised learning (SSL), wav2vec 2.0 [30] has gained significant attention in the field of ASR. To adhere to a widely accepted speech representation model, we utilize a well-trained wav2vec 2.0 base-based ASR model, as used in the original Sidecar paper [8]. Additionally, to validate the feasibility and generality of the Sidecar approach, we also employ a data2vec 2.0 base-based ASR model as another backbone for comparison [27]. The two models differ primarily in two ways: (1) they are pre-trained using different protocols, resulting in variations in the encoded representations, and (2) data2vec 2.0 biases the query-key attention scores with a penalty proportional to their distance.

Both ASR models are well-trained and comprise a CNN feature extractor, a Transformer encoder, and a fully-connected layer as the decoder. Specifically, the model takes waveform as input and extracts acoustic features using a seven-layer CNN feature extractor. The extracted features will be fed into the 12-layer Transformer encoder to generate high-level representations. Following the paradigm outlined in [30] and [27], we use only a fully-connected layer as the decoder for letter-level prediction. We directly utilize fairseq's official released model parameters [31], and denote them as *W2V-CTC* and *D2V-CTC* throughout subsequent sections.

2.2. Sidecar separator

Enlightened by the findings that the ASR encoder captures more acoustic information in its lower layers and more linguistic information in the upper layers [9, 10], a recent study proposes using a Sidecar separator to address multi-talker speech recog-
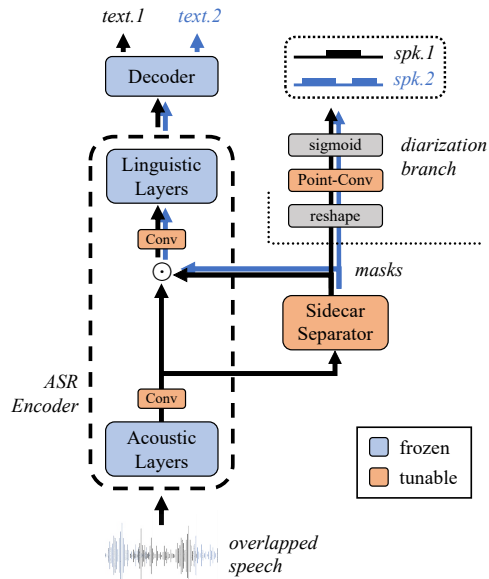

Figure 1: *The proposed strategy plugs a Sidecar separator with a diarization branch into a frozen well-trained single-talker ASR model, enabling it unified modeling for multi-talker overlapped speech recognition and diarization. The Sidecar is with Conv-TasNet-like architecture.*

nition, drawing on methodologies in speech separation [8].

The Sidecar separator is a temporal convolutional network that comprises stacked 1-D dilated convolutional blocks similar to Conv-TasNet [32]. This design allows the Sidecar to model long-term dependencies of acoustic embeddings while maintaining a small size. As shown in Figure 1, a 3-kernel-size 1-D convolutional layer is employed on each side of the Sidecar to filter the input-mixed and output-separated embeddings. Following the design proposed in [8], we plug the Sidecar between the second and the third encoder layers as a compromise in semantics.

During the forward process, the mixed speech embedding generated by the preceding layer is filtered through a convolutional layer and then fed into the Sidecar to synthesize speaker-dependent masks. These masks are used to element-wise multiply the filtered mixed speech embedding, and the resulting product is further adjusted with another convolutional layer to obtain separated embeddings. These embeddings, corresponding to different speakers, are concatenated onto the batch dimension for parallel processing and transcription into text.

2.3. The diarization branch and processes for diarization

Furthermore, we incorporate a diarization branch into the Sidecar to enable the unified modeling of speech recognition and diarization. As illustrated in Figure 1, the main component of the branch is a point-wise 2-D convolutional layer.

In the forward process, the speaker-dependent masks, as shown in Figure 1, generated by the Sidecar possess a tensor shape of $(B \times S, C, T)$ where B denotes batch size, S denotes the number of speakers, C denotes the number of channels, and T denotes time frames. Each time frame spans a duration of 20 ms. Within the diarization branch, these masks are first reshaped to (B, S, C, T) before being transposed to (B, C, S, T). The reshaped-and-transposed masks then go through a point-wise 2-D convolutional layer, which is the only trainable layer in the branch having C parameters, to generate a tensor with a

shape of $(B, 1, S, T)$. After squeezing on the second dimension and applying a sigmoid activation function, frame-wise predictions for each speaker's speech activities D are synthesized with a shape of (B, S, T). For every speaker s in S and for each time frame t in T, if the element value is greater than 0.5, we consider that speaker s was activated on the time frame t. This will yield the results for diarization.

During the training phase on LibriMix and LibriSpeechMix datasets, the model is fed with complete utterances. However, during the adaptation and inference for diarization on CALL-HOME (Section 4.2), we segment the utterances to ensure the alignment with real-world diarization scenarios and guarantee its practicality. As depicted in Figure 2, we divide each utterance into several 30-second segments that share a common 15-second interval between every two adjacent segments. For shared parts, we calculate Euclidean distance for different speaker permutations between segment tensors D and select one with minimum distance to modify the speaker arrangement in subsequent segments. Afterward, we average the element values of adjacent segments' shared parts. Note that this segmenting-and-permuting process is not utilized in the experiments mentioned in Section 4.1.

2.4. Training and adaptation objectives

During training on LibriMix and LibriSpeechMix datasets, both CTC loss and diarization loss require a permutation for the speaker order to be assigned to address the label ambiguity issue [28]. To explicitly construct the inter-dependence between the two tasks, the permutation is determined by permutation invariant training (PIT) based on CTC loss, and then is assigned for diarization loss. The diarization loss is to calculate the mean squared error (MSE) between the predicted speaker activities D and the diarization ground truth. At last, the final objective function is the sum of PIT-CTC loss and corresponding diarization loss multiplied by a coefficient λ.

However, when we adapt the model for CALLHOME, we solely employ diarization loss and determine speaker permutation relying on the strategy outlined in Section 2.3.

3. Experimental Setup

3.1. Datasets

The experiments are performed on two benchmark datasets for multi-talker ASR (LibriMix [33] and LibriSpeechMix [4]), and the two-speaker subset of a real-world dataset for diarization (CALLHOME). Although LibriMix and LibriSpeechMix datasets were not specifically designed for the diarization task, we also present diarization results on them.

LibriSpeechMix. The utterances are simulated with the mixtures of two or three speakers from LibriSpeech. Only standard official dev and test sets are published. Our training set is homemade from the 960-hour LibriSpeech training dataset (LS-960) referring to the protocol established in [4]. LibriSpeechMix randomly samples a delay time for the second and the third utterances, so the mixture is partially overlapping.

LibriMix. The dataset simulates audio mixtures using a combination of two or three speakers sourced from the LibriSpeech-clean corpus. We focus on its two-speaker-clean subset *Libri2Mix-clean* and three-speaker-clean subset *Libri3Mix-clean*. The mixtures are made in a left-aligned style. Thus, the shorter source speech will be entirely overlapped by the longer one from the start, which challenges the model more than LibriSpeechMix in separating overlaps.

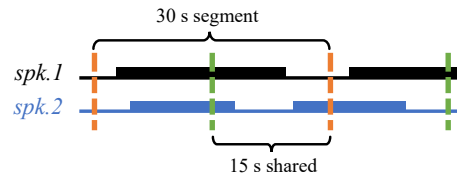**CALLHOME**. We evaluate the proposed method on the di-

Figure 2: *The pre-process for diarization on CALLHOME. The interval between the green and orange dashed lines is the shared interval for the two segments.*

arization task with CALLHOME, which is a benchmark dataset consisting of spontaneous multilingual telephone conversations, as one part of the 2000 NIST Speaker Recognition Evaluation (LDC2001S97). We take its two-speaker subset and split it into an adaptation set of 155 recordings and a test set of 148 recordings, following the same partition protocol with EEND [11, 12] and Kaldi[1] [34]. The average duration is 73.1 seconds.

3.2. Model settings

Well-trained single-talker ASR model. We utilize the well-trained W2V-CTC and D2V-CTC as the backbone, respectively. To ensure consistency with [30] and [27], we directly employ the official released model weights by fairseq[2] [31]. Both models are pre-trained on unlabeled LS-960 data and subsequently fine-tuned on labeled LS-960 using CTC loss. The resulting well-trained models are then frozen for use in our experiments.
Sidecar separator. Drawing inspiration from [32], the Sidecar separator implements a sequence of K temporal convolutional blocks with dilation rates spanning from 1 to 2^{K-1}, repeated up to K times. To align with the protocol in [8], we set $K = 8$ and $R = 3$, remove skip-connection paths within the convolutional blocks, and substitute the final sigmoid activation with ReLU. The Sidecar has 128 bottleneck channels and 768 input/output channels. The Sidecar is plugged between the second and the third transformer layers, which mirrors that of [8].
Diarization branch. As shown in Figure 1, the speaker-dependent masks predicted by Sidecar will be fed into the diarization branch. The main component of the branch is a point-wise 2-D convolutional layer with an input-channel size of 756 and output-channel size of 1, featuring a kernel size of $(1, 1)$ with stride 1. As such, this branch only requires an additional 768 parameters. As is typical, we use a collar of 250 ms when evaluating the DER performance [11, 12].
Training settings. With W2V-CTC or D2V-CTC frozen, the models only have 8.7 M trainable parameters (8.4% of all parameters) for the two-speaker experiments and 8.8 M trainable parameters (8.5% of all parameters) for the three-speaker experiments. We set the coefficient of diarization loss λ to 0.01. Adhering to the settings in [8], we optimize the proposed models using a 2e-4 learning rate with a three-stage schedule and Adam optimizer, for at most 100 k updates. It takes about 7 hours for two-speaker models and 9 hours for three-speaker models with 8 NVIDIA V100 GPUs, thanks to Sidecar's small size and the ejection start provided by the well-trained ASR model.

In the following, we denote the proposed models as *W2V-Sidecar*, *W2V-Sidecar-DB*, *D2V-Sidecar*, *D2V-Sidecar-DB*, where "*-DB*" denotes "with diarization branch". Note that *W2V-Sidecar* is identical to the implementation in [8], serving as the baseline in this work. Permutations with minimum errors are used to compute word error rates (WERs).

---

[1]https://github.com/kaldi-asr/kaldi/tree/master/egs/callhome_diarization/v2
[2]https://github.com/facebookresearch/fairseq

4. Results and Discussion

4.1. Results on LibriMix and LibriSpeechMix datasets

The models (b) and (d) in Table 1 are optimized with CTC and diarization loss, enabling the unified modeling of multi-talker ASR and diarization.

ASR. The ASR results of the four models on two- and three-speaker LibriMix and LibriSpeechMix datasets are presented in Table 1. Aligning with our previous hypothesis, the enhanced model (b) with a diarization branch consistently outperforms the baseline (a) [8] across all four datasets, benefiting from the inter-dependent and complementarity of the two tasks. Additionally, models utilizing D2V as a backbone generally outperform those with W2V. We attribute the boost in performance of models (c) and (d) to the better representations learned in its pre-training phase as illustrated in [27]. The diarization branch's performance enhancement on the LibriSpeechMix-2pk is relatively modest, and we contend that this is due to the dataset being relatively simple for ASR owing to its lower two-speaker overlap rate. This study achieves the state-of-the-art performances on Libri2Mix, and it is the first to report ASR results on Libri3Mix in the field. However, WERs on Libri3Mix remains high due to shorter source speeches being completely overlapped by longer ones from the outset, rendering it more challenging for multi-talker ASR.

Diarization. Although LibriMix and LibriSpeechMix datasets were not created for diarization purposes, we include the diarization results on them in Table 2 to offer a more comprehensive perspective. As a similar trend to Table 1, D2V backbone achieves better performance on the diarization task compared to W2V. Note that while left-aligned-style generated LibriMix dataset poses greater challenges for ASR than LibriSpeechMix, it proves easier for the diarization task since it only requires predicting speaker activity timestamps on the right side.

4.2. Diarization results on CALLHOME dataset

To demonstrate its practicality, we conducted the evaluation of the proposed method on the real-world CALLHOME dataset under realistic settings. Specifically, we adapt W2V-Sidecar-DB and D2V-Sidecar-DB, which have been trained on Libri2Mix, to the two-speaker subset of CALLHOME. During adaptation and inference, the segmenting-and-permuting process is employed to align with real-world diarization scenes, as discussed in Section 2.3. As shown in Table 3, in comparison to EEND models that are carefully designed for diarization and trained on datasets tailored for this purpose, our method delivers satisfactory performance with just 8.7 M trainable parameters and a few adaptation steps, demonstrating the effectiveness and flexibility of our approach with limited resources.

We observed that the proposed method can achieve better or comparable performance on MI and FA, which are metrics related to voice activity detection, but falls behind in CF. We argue that this is because the used Libri2Mix dataset is clean and designed for ASR purpose with only 1,172 speakers, far fewer than the 5,743 speakers of the carefully crafted dataset used by EEND. We anticipate significant improvement by training it with data simulated specifically for diarization purposes as done by [11, 12].

4.3. Limitations and future work

While this work is innovative in unifying the modeling of speech recognition and diarization, several limitations remain. Firstly, due to considerations of simplicity and comparability with previous work [8], the model's performance on diariza-

Table 1: *ASR performance on the test sets of LibriMix and LibriSpeechMix. Evaluated by WER (%). "-DB" refers to "with diarization branch"*

System	LibriMix		LibriSpeechMix	
	2spk	3spk	2spk	3spk
(a) W2V-Sidecar[†] [8]	10.36	35.22	7.56	13.87
(b) W2V-Sidecar-DB	9.88	34.38	7.53	12.93
(c) D2V-Sidecar	10.11	34.84	7.61	12.56
(d) D2V-Sidecar-DB	**9.69**	**33.91**	**7.49**	**11.94**

[†] Serving as the baseline in this work.

Table 2: *Diarization performance on the test sets of LibriMix and LibriSpeechMix. Evaluated by DER (%).*

System	LibriMix		LibriSpeechMix	
	2spk	3spk	2spk	3spk
W2V-Sidecar-DB	0.97	2.35	2.20	3.65
D2V-Sidecar-DB	0.91	2.14	2.12	3.47

Table 3: *Detailed diarization results on CALLHOME, evaluated by DER (%), which is the sum of misses (MI), false alarms (FA), and speaker confusions (CF) errors.*

System	MI	FA	CF	DER
BLSTM-EEND [11]	-	-	-	23.07
SA-EEND [12]	6.68	2.40	1.68	**10.66**
W2V-Sidecar-DB (ours)	6.26	3.56	5.07	14.89
D2V-Sidecar-DB (ours)	6.18	3.80	4.64	14.62

tion is acceptable but restricted by its training strategy. However, access to more suitable datasets and training schemes may yield improved performance. Secondly, the system requires pre-defining the maximum number of speakers addressed, which could potentially be resolved by maintaining a speaker embedding bank in the future. Lastly, the current model does not tackle the "who spoke when and what" issue. Nevertheless, we believe this work has pointed the direction toward a potential solution to it.

5. Conclusion

A recent study proposed a low-cost approach for converting a single-talker ASR system to a multi-talker one, by plugging a Sidecar separator into a fixed well-trained common ASR model. Extending on this approach, we incorporate a diarization branch into the Sidecar with only 768 additional parameters, allowing for unified modeling of both multi-talker overlapped speech recognition and diarization tasks with a low training cost.

With very few parameters (8.7 M for the two-speaker model, and 8.8 M for the three-speaker model) requiring tuning, the proposed approach outperforms the original Sidecar scheme on ASR tasks for LibriMix and LibriSpeechMix datasets. Furthermore, without sophisticated customization on the diarization task, the proposed method achieves potential diarization results on the two-speaker subset of the real-world CALLHOME dataset, with only a few adaptation steps.

6. Acknowledgements

7. References

[1] J. Li, "Recent advances in end-to-end automatic speech recognition," *APSIPA Transactions on Signal and Information Processing*, vol. 11, no. 1, 2022.

[2] S. Settle, J. L. Roux, T. Hori, S. Watanabe, and J. R. Hershey, "End-to-end multi-speaker speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4819–4823.

[3] S. Li, B. Ouyang, F. Tong, D. Liao, L. Li, and Q. Hong, "Real-time end-to-end monaural multi-speaker speech recognition," in *Proc. Interspeech 2021*, pp. 3750–3754.

[4] N. Kanda, Y. Gaur, X. Wang, Z. Meng, and T. Yoshioka, "Serialized output training for end-to-end overlapped speech recognition," in *Proc. Interspeech 2020*, pp. 2797–2801.

[5] L. Lu, N. Kanda, J. Li, and Y. Gong, "Streaming end-to-end multi-talker speech recognition," *IEEE Signal Processing Letters*, vol. 28, pp. 803–807, 2021.

[6] N. Kanda, J. Wu, Y. Wu, X. Xiao, Z. Meng, X. Wang, Y. Gaur, Z. Chen, J. Li, and T. Yoshioka, "Streaming multi-talker ASR with token-level serialized output training," in *Proc. Interspeech 2022*, pp. 3774–3778.

[7] L. Lu, N. Kanda, J. Li, and Y. Gong, "Streaming multi-talker speech recognition with joint speaker identification," in *Proc. Interspeech 2021*, pp. 1782–1786.

[8] L. Meng, J. Kang, M. Cui, Y. Wang, X. Wu, and H. Meng, "A Sidecar separator can convert a single-talker speech recognition system to a multi-talker one," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[9] A. Pasad, J.-C. Chou, and K. Livescu, "Layer-wise analysis of a self-supervised speech representation model," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2021, pp. 914–921.

[10] K. Shim, J. Choi, and W. Sung, "Understanding the role of self attention for efficient speech recognition," in *International Conference on Learning Representations*, 2022.

[11] Y. Fujita, N. Kanda, S. Horiguchi, K. Nagamatsu, and S. Watanabe, "End-to-end neural speaker diarization with permutation-free objectives," in *Proc. Interspeech 2019*, pp. 4300–4304.

[12] Y. Fujita, N. Kanda, S. Horiguchi, Y. Xue, K. Nagamatsu, and S. Watanabe, "End-to-end neural speaker diarization with self-attention," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 296–303.

[13] Z. Huang, S. Watanabe, Y. Fujita, P. García, Y. Shao, D. Povey, and S. Khudanpur, "Speaker diarization with region proposal network," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6514–6518.

[14] S. Horiguchi, Y. Fujita, S. Watanabe, Y. Xue, and K. Nagamatsu, "End-to-end speaker diarization for an unknown number of speakers with encoder-decoder based attractors," in *Proc. Interspeech 2020*, pp. 269–273.

[15] S. H. Shum, N. Dehak, R. Dehak, and J. R. Glass, "Unsupervised methods for speaker diarization: An integrated and iterative approach," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 10, pp. 2015–2028, 2013.

[16] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker diarization using deep neural network embeddings," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 4930–4934.

[17] N. Zheng, N. Li, X. Wu, L. Meng, J. Kang, H. Wu, C. Weng, D. Su, and H. Meng, "The CUHK-TENCENT speaker diarization system for the ICASSP 2022 multi-channel multi-party meeting transcription challenge," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 9161–9165.

[18] T. J. Park, N. Kanda, D. Dimitriadis, K. J. Han, S. Watanabe, and S. Narayanan, "A review of speaker diarization: Recent advances with deep learning," *Computer Speech & Language*, vol. 72, p. 101317, 2022.

[19] L. E. Shafey, H. Soltau, and I. Shafran, "Joint speech recognition and speaker diarization via sequence transduction," in *Proc. Interspeech 2019*, pp. 396–400.

[20] H. H. Mao, S. Li, J. McAuley, and G. W. Cottrell, "Speech recognition and multi-speaker diarization of long conversations," in *Proc. Interspeech 2020*, pp. 691–695.

[21] N. Kanda, Y. Gaur, X. Wang, Z. Meng, Z. Chen, T. Zhou, and T. Yoshioka, "Joint speaker counting, speech recognition, and speaker identification for overlapped speech of any number of speakers," in *Proc. Interspeech 2020*, pp. 36–40.

[22] N. Kanda, J. Wu, Y. Wu, X. Xiao, Z. Meng, X. Wang, Y. Gaur, Z. Chen, J. Li, and T. Yoshioka, "Streaming speaker-attributed ASR with token-level speaker embeddings," in *Proc. Interspeech 2022*, pp. 521–525.

[23] N. Kanda, S. Horiguchi, Y. Fujita, Y. Xue, K. Nagamatsu, and S. Watanabe, "Simultaneous speech recognition and speaker diarization for monaural dialogue recordings with target-speaker acoustic models," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 31–38.

[24] N. Kanda, X. Xiao, Y. Gaur, X. Wang, Z. Meng, Z. Chen, and T. Yoshioka, "Transcribe-to-diarize: Neural speaker diarization for unlimited number of speakers using end-to-end speaker-attributed asr," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 8082–8086.

[25] Z. Huang, M. Delcroix, L. P. Garcia, S. Watanabe, D. Raj, and S. Khudanpur, "Joint speaker diarization and speech recognition based on region proposal networks," *Computer Speech & Language*, vol. 72, p. 101316, 2022.

[26] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "Data2vec: A general framework for self-supervised learning in speech, vision and language," in *International Conference on Machine Learning*, 2022, pp. 1298–1312.

[27] A. Baevski, A. Babu, W.-N. Hsu, and M. Auli, "Efficient self-supervised learning with contextualized target representations for vision, speech and language," *arXiv preprint arXiv:2212.07525*, 2022.

[28] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 241–245.

[29] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[30] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Advances in neural information processing systems*, vol. 33, 2020, pp. 12449–12460.

[31] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, "fairseq: A fast, extensible toolkit for sequence modeling," in *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019.

[32] 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*, 2019.

[33] J. Cosentino, M. Pariente, S. Cornell, A. Deleforge, and E. Vincent, "LibriMix: An open-source dataset for generalizable speech separation," *arXiv preprint arXiv:2005.11262*, 2020.

[34] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The Kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, 2011.