MULTI-TURN RNN-T FOR STREAMING RECOGNITION OF MULTI-PARTY SPEECH

Ilya Sklyar[1*], *Anna Piunova*[1*], *Xianrui Zheng*[2+], *Yulan Liu*[†]

[1]Amazon Alexa, [2]University of Cambridge

ilsklyar@amazon.com piunova@amazon.com xz396@eng.cam.ac.uk

ABSTRACT

Automatic speech recognition (ASR) of single channel far-field recordings with an unknown number of speakers is traditionally tackled by cascaded modules. Recent research shows that end-to-end (E2E) multi-speaker ASR models can achieve superior recognition accuracy compared to modular systems. However, these models do not ensure real-time applicability due to their dependency on full audio context. This work takes real-time applicability as the first priority in model design and addresses a few challenges in previous work on multi-speaker recurrent neural network transducer (MS-RNN-T). First, we introduce on-the-fly overlapping speech simulation during training, yielding 14% relative word error rate (WER) improvement on LibriSpeechMix test set. Second, we propose a novel multi-turn RNN-T (MT-RNN-T) model with an overlap-based target arrangement strategy that generalizes to an arbitrary number of speakers without changes in the model architecture. We investigate the impact of the maximum number of speakers seen during training on MT-RNN-T performance on LibriCSS test set, and report 28% relative WER improvement over the two-speaker MS-RNN-T. Third, we experiment with a rich transcription strategy for joint recognition and segmentation of multi-party speech. Through an in-depth analysis, we discuss potential pitfalls of the proposed system as well as promising future research directions.

Index Terms— streaming multi-speaker speech recognition, overlapped speech, recurrent neural network transducer, multi-turn

1. INTRODUCTION

Recognizing multi-party speech from an arbitrary number of speakers with naturally occurring speech overlap in a far-field environment is a challenging research problem that has been extensively studied for years [1, 2, 3, 4]. This problem is frequently referred to as the "cocktail party problem". Its solution holds the key to the next era of speech technology, and lays the foundation for natural human-device voice interactions. In such interactions, the speech agent on smart device is expected to navigate through multi-party conversations and provide the support in need. To allow real-time human-device interactions, the ASR system needs to provide transcripts in a streaming fashion for all speakers in the conversation. For better privacy protection, the ASR system design needs to accommodate local operation on device only, which means the model should be light-weight and robust against acoustic front-end limitations. All these factors point to the importance of streaming speech recognition of an arbitrary number of speakers with potential speech overlaps using single distant microphone only.

Traditionally, multi-speaker ASR was approached with cascaded modules such as speech separation and speech recognition [5, 6, 7, 8, 9]. Due to the system complexity and technical limitation in some modules, such systems are more suitable for off-line applications where the full audio recording is available before transcription task starts. In past years, research progress has been made in training joint models that optimize multi-speaker ASR performance directly [10, 11, 12, 13, 14, 15]. Recently, [16] proposed joint multi-speaker speech recognition and speaker change detection for any number of speakers via serialized output training (SOT), and demonstrated its effectiveness on simulated multi-speaker test set of LibriSpeechMix. Later [17] showed that SOT is also effective on real multi-speaker meeting corpus, and subsequent works [18, 19, 20, 21] extended SOT into speaker-attributed ASR that can transcribe "who spoke what" in multi-speaker conversations with one integrated model.

In parallel, recent research also started to look at maintaining the same performance under streaming conditions. Two conceptually similar streaming multi-speaker ASR models based on a recurrent neural network transducer (RNN-T) [22], i.e. MS-RNN-T [23] and streaming unmixing and recognition transducer (SURT) [24], were concurrently proposed to allow time-synchronous decoding of partially overlapping speech, and [24] was later extended with speaker identification in [25]. [23] conducted evaluations of MS-RNN-T on single-speaker LibriSpeech and two-speaker LibriSpeechMix. It showed that MS-RNN-T achieves on-par performance with off-line multi-speaker ASR systems on two-speaker test data and also improves robustness of the system on noisy single-speaker test-other partition of LibriSpeech.

This work builds on [23], with the following contributions. First, we show the advantage of on-the-fly overlapping speech simulation over fixed pre-simulated data for training, leading to a new low WER on two-speaker partition of LibriSpeechMix with a streamable model architecture. Second, we address the main limitation of MS-RNN-T, i.e. the hard restriction on the maximum number of speakers in the audio. Assuming there are up to two speakers overlapping at a time, the proposed MT-RNN-T model is capable to recognize speech from an arbitrary number of speakers. Additionally, we make the first step towards streaming rich transcription of multi-party speech by introducing segmentation tag. Last but not least, we analyze performance in-depth using proposed evaluation metrics, optimal reference combination (ORC) WER and turn counting accuracy, and discuss potential future work.

2. PRIOR WORK

MS-RNN-T [23] extends the standard RNN-T [22] to overlapping speech recognition with multiple output channels $N = S$, where S is the number of speakers in the audio. The encoder of MS-RNN-T has a modular structure containing a mixture encoder (MixEnc), N separation encoders (SepEnc_n) for each output channel $n \in \{1, ..., N\}$ and a recognition encoder (RecEnc) with shared parameters between output channels. The encoder takes acoustic features

[*] These authors have contributed equally. [+] The contribution was fully conducted during internship in Amazon. [†] Work was done while in Amazon.

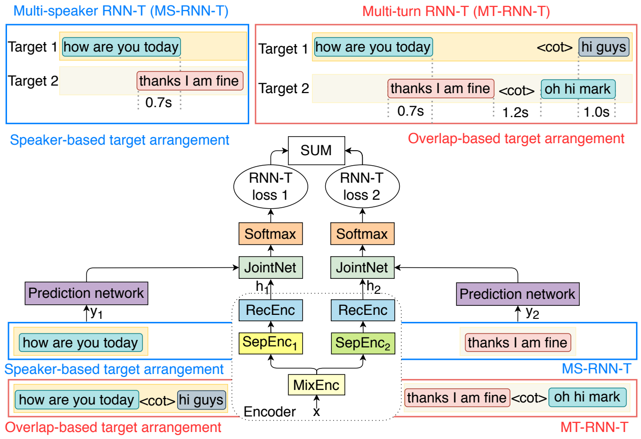

Fig. 1: Baseline MS-RNN-T with speaker-based target arrangement from [23], and proposed MT-RNN-T with overlap-based target arrangement with optional <cot> tag for speech segmentation. Model blocks with the same colour have tied parameters, transcripts in the colour-matched boxes belong to the same speaker.

$\mathbf{x}$ as input and produces high-level disentangled acoustic representations $\mathbf{h}_n$ as output, as described mathematically in Eq. 1 and visualized in the corresponding block in Fig. 1.

$$\mathbf{h}_n = \text{RecEnc}(\text{SepEnc}_n(\text{MixEnc}(\mathbf{x}))) \tag{1}$$

[23] proposed two ways to associate $\mathbf{h}_n$ with prediction network outputs for each label sequence $\mathbf{y}_n$: deterministic assignment training (DAT) and permutation invariant training (PIT). DAT forces the model to learn to associate its output with the speaker order in the audio. For the two-speaker case, the first separation encoder learns to focus on the leading speaker, and the second on the follow-up speaker if it exists. DAT computes RNN-T loss N times, as described in Eq. 2 and depicted on Fig. 1. In contrast, PIT allows the separation encoder to flexibly match either of the speaker, as long as it minimizes the loss function during training. Therefore, PIT carries out N^2 RNN-T loss computations that are required to find an optimal permutation π of speaker n from the set of permutations $\mathcal{P}$ (Eq. 3).

$$\mathcal{L}_{DAT} = -\sum_n \log P(\mathbf{y}_n|\mathbf{h}_n) \tag{2}$$

$$\mathcal{L}_{PIT} = \min_{\pi\in\mathcal{P}} -\sum_n \log P(\mathbf{y}_n|\mathbf{h}_{\pi(n)}) \tag{3}$$

While [23] showed that PIT is better than DAT at recognizing overlapping speech, PIT generally has challenges in scaling up to a large number of speakers due to its $O(N!)$ complexity.

3. TECHNICAL APPROACHES

3.1. On-the-fly multi-speaker speech audio simulation

Previous work [17] showed the benefit of pre-training on the artificial multi-speaker data simulated on-the-fly in comparison to the training on real multi-speaker data from scratch. In this work we benchmark on-the-fly simulation against training on the fixed simulated data in close-talking conditions. We demonstrate that due to the increased variety of overlapping speech scenarious seen during training it improves generalization to unseen multi-speaker audio.

To simulate the audio mixture, we select S utterances randomly from a pool of single-speaker utterances. For each follow-up utterance, we sample a random time delay from a specified delay range. Before mixing, a reference speaker is randomly determined and the speech energy of other speakers is normalized to achieve an energy ratio randomly sampled from a specified range. To simulate far-field recording scenario we convolve each source audio with an acoustic impulse response (AIR) before padding with the sampled delay.

3.2. Overlap-based target arrangement and MT-RNN-T

Along with the audio simulation of multi-speaker speech, the transcripts should also be combined on-the-fly accordingly. For two-speaker MS-RNN-T mixture, when the audio contains two overlapping speakers, the two reference transcripts are simply assigned to two targets. When there is no overlapped speech, there are different strategies to assign the reference. Previous work [23] adopted speaker-based target arrangement (Fig. 1 blue), which always assigns reference transcripts to different targets. This strategy encourages the model to separate speaker by dedicating each output for one unique speaker, but it does not scale up to a large number of speakers in audio. Here we propose a different strategy, overlap-based target arrangement, which allows to build two targets for two model outputs by combining references into two targets in the order of their appearance in the audio. Overlap-based arrangement concatenates utterances together if there is no overlap between them and only incurs target switch at speech overlap between two consecutive utterances, as shown on Fig. 1 (red). This simple change in the training data pipeline allows us to train models on data with more speakers S and speaker turns U than the number of output channels N: $U \geq S > N$. Therefore, we refer to models trained with overlap-based target arrangement as multi-turn RNN-T (MT-RNN-T).

We observed that overlap-based target switch is similar to the recent graph-PIT work [26], though our work was completed fully in parallel and independently. The target assignment is equivalent to graph-PIT but we do not permute different combinations of reference targets and model hypothesis for loss calculation. This effectively leads to complexity reduction from $O(N^U)$ in graph-PIT to $O(N)$ in this work.

3.3. Rich transcriptions with speech segmentation

Overlap-based target arrangement in MT-RNN-T makes speaker tracking more difficult since multiple speakers can be assigned to the same output channel. For downstream applications, speaker tracking in the multi-speaker ASR system is important to provide not only "what" was said, but also "who" said what. As the first step towards such rich transcriptions, in this work we introduce segmentation tags to indicate potential point of speaker turn change, so that future work on speaker labelling could be built on top. To emphasize this conceptual difference from other research work based on speaker change detection [16, 19], we refer to each segment as a "turn", without necessarily insisting the semantic correctness of this term. We augment model vocabulary with a "change-of-turn" (<cot>) token which is inserted between two consecutive turns in the same target, as depicted on Fig. 1.

4. EXPERIMENTS

4.1. Experiments on LibriSpeechMix

We perform our first set of experiments on LibriSpeechMix to verify the effectiveness of on-the-fly data simulation setup on a simpler task with limited number of speakers.

Table 1: Benchmarking on-the-fly-simulation with MS-RNN-T model variants against other streaming methods on LibriSpeechMix.

Model	On-the-fly simulation	WER [%]
SURT [24]	✗	10.8
SURT [25]	✗	10.3
DAT-MS-RNN-T [23]	✗	11.0
DAT-MS-RNN-T (this work)	✓	9.2
PIT-MS-RNN-T [23]	✗	10.2
PIT-MS-RNN-T (this work)	✓	8.8

Task description – LibriSpeechMix is a simulated dataset initially proposed in [16] that contains artificially mixed utterances from LibriSpeech corpus [27]. We use 2-speaker dev and test partitions of this dataset with a single turn for each speaker in the utterance. Delay for the second speaker is randomly sampled with the constraint that each audio mixture has an overlapping segment. The original training dataset of LibriSpeechMix follows a similar simulation strategy with an additional constraint on a minimal delay of 0.5 sec between start times of speaker turns. It contains ∼1.5k hours of simulated data, and we substitute it with the proposed on-the-fly data simulation pipeline in the experiments presented below.

Training setup – The model topology of MS-RNN-T is based on the one established in [23]. We use 2 LSTM layers in each recurrent module of the architecture (mixture encoder, 2 separation encoders, recognition encoder, prediction network) with 1024 units in each layer. Output layers in the recognition encoder and the prediction network have 640 units. The joint network has a single feed-forward layer with 512 units. The output softmax layer has dimensionality of 2501 which corresponds to the blank label and 2500 wordpieces that represent the most likely subword segmentation from a unigram word piece model [28]. Acoustic features are 64-dimensional log-mel filterbanks with a frame shift of 10ms which are stacked and downsampled by a factor of 3. We use SpecAugment with Libri-FullAdapt policy [29] for feature augmentation. We use the Adam algorithm [30] with the warm-up, hold and decay schedule proposed in [31] for the optimization of all models. We select the best model checkpoint based on performance on the development set.

Similar to the previous work [23] we pre-train MS-RNN-T model with a single separation encoder on the LibriSpeech dataset before training with both separation encoders on the simulated multi-speaker data. During on-the-fly simulation we randomly sample two utterances from different speakers in the pool of LibriSpeech data. Consistent with the simulation configuration of the original LibriSpeechMix training corpus, time delay for the second speaker is sampled uniformly from the range: $(0.5, \text{len}(utt_1))$, while original signal energy is remained intact.

Results – For the evaluation we used the same metric as in [23], i.e. optimal edit distance WER (OED WER), that is also aligned with [15, 16]. Table 1 shows the benefit of on-the-fly simulation on MS-RNN-T models based on both DAT and PIT. We compare their performance with the corresponding models trained on the fixed simulated dataset in [23], external SURT proposed in [24] and its improved variant from [25]. All models in this comparison have streamable architectures and comparable number of parameters (∼81M). On-the-fly simulation brings substantial improvements for both DAT and PIT training variants of MS-RNN-T with 16% and 14% relative WER reductions, respectively. It is noteworthy that relative WER difference between DAT and PIT training approaches diminish from 7% to 4% when models are exposed to more diverse multi-speaker data generated on-the-fly. MS-RNN-T with permutation invariant training (PIT-MS-RNN-T) with on-the-fly-simulation achieves the new state-of-the-art performance on this benchmark among all streaming methods and outperforms the best external SURT model by 14% relative WER.

4.2. Experiments on LibriCSS

We perform our second set of experiments on LibriCSS dataset where we benchmark MT-RNN-T model variants on partially overlapped multi-turn speech from more than 2 speakers.

Task description – LibriCSS was originally proposed in [32] for continuous speech separation. It contains 10 far-field audio recordings with LibriSpeech utterances played back in a room to simulate meetings with 8 speakers. The full LibriCSS evaluation data is divided into 6 partitions. 2 partitions exclude overlapped speech, with either short (0S) or long (0L) silence gaps between speaker turns. The rest 4 partitions cover different overlap ratios, from 10% to 40%: OV10, OV20, OV30 and OV40. We perform segmentation of the original one-hour long LibriCSS sessions into utterance group segments using oracle silence boundary information as described in [19]. It ensures the existence of both single-turn (single utterance) and multi-turn segments in the evaluation data. This type of multi-speaker ASR model benchmarking is also known as utterance group evaluation in the literature [17], and it is important to diffentiate it from utterance-wise and continuous input evaluation protocols discussed in [32]. We use Session 0 of the dataset as a development set to tune the hyper-parameter word-reward [33, 34] and to select the best checkpoint in a grid search fashion. The remaining Sessions 1-9 are used to report performance.

Training setup – We closely follow training setup used in LibriSpeechMix experiments (Section 4.1) for experiments on LibriCSS, with a few important differences. First, we introduce layer normalization [35] in each LSTM layer of the model architecture, which we find beneficial to improve convergence on more challenging training data. Second, we extend on-the-fly simulation to support far-field conditions with multi-party speech. We sample random number of utterances uniformly from the range $\{1, \dots, S\}$. We do not impose the constraint on all utterances to come from different speakers, however the overlap between segments uttered by the same speaker is not allowed. Each utterance s is scaled and convolved with AIR before adding to the mixture. Time delay for each subsequent overlapping utterance utt_{s+1} is sampled from the range $(0.5, \text{len}(utt_s))$, which ensures overlapped speech segment existence in all simulated multi-speaker utterances. Energy ratio between the reference speaker and each other speaker is sampled uniformly from the range -5 dB to 5 dB. We filter out simulated audio if its length exceeds 30 seconds to avoid out-of-memory errors in the RNN-T loss. This hardware limitation makes it inefficient to experiment with large maximum number of speakers S in the simulation, therefore we did not go beyond $S = 5$ in the experiments.

Evaluation metric – In our early experiments we found that the previously used OED WER metric could not well represent the recognition accuracy on LibriCSS test data due to the difficulty in predicting from which output thread the model will output the hypothesis for a specific region of audio. Therefore, to provide an accurate picture of the recognition performance, we propose a new evaluation metric where the reference arrangement is optimized based on model

Table 2: MT-RNN-T model benchmarking on LibriCSS with performance reported in optimal reference combination WER [%].

Model ID	Model type	Max #spk in training	<cot>	0L	0S	OV10	OV20	OV30	OV40	full
0	MS-RNN-T	2	✗	16.1	15.8	25.0	35.2	46.0	48.0	32.8
1	MT-RNN-T	3	✗	14.9	15.4	23.3	31.0	35.0	39.0	27.8
2	MT-RNN-T	4	✗	15.1	15.3	20.6	27.2	34.0	36.8	26.0
3	MT-RNN-T	5	✗	14.8	14.5	18.0	25.8	30.3	32.3	23.6
4	MT-RNN-T	5	✓	14.7	14.8	20.7	25.3	33.2	36.4	25.3

output to provide the lowest WER readings, i.e. optimal reference combination WER (ORC WER). Assuming that there are U reference transcripts for the audio, the number of transcripts that can be assigned to the first model output varies from 0 to U, while the rest is assigned to the second output. Then ORC WER can be efficiently computed as follows. First, we generate all C_U^k possible reference combinations that assign reference transcripts to two model outputs, where $k \in \{0, \ldots, U\}$ is the number of reference transcripts to assign to the first model output. Then, we sort the references by start time and concatenate references within each generated combination. For each combination, we compute the overall WER of two model outputs against two concatenated reference combinations, then report the lowest WER among all combinations. This approach factors out the reference-hypothesis pairing errors from the actual word recognition errors. In the evaluation, we left out the longest utterance (with 24 turns) obtained after segmentation from ORC WER scoring as we found it computationally infeasible to search over all possible reference combinations. In addition, segmentation token <cot> is excluded from hypotheses for scoring.

Results – Evaluation results on LibriCSS dataset are presented in Table 2. We use MS-RNN-T trained with on-the-fly simulated audio containing up to 2 speakers (model 0) as the baseline. Since switching from MS-RNN-T to MT-RNN-T allows us to train on simulated data with more than 2 speakers, in models 1-3 we incrementally increase the maximum number of speakers in training from 2 to 5. We observe that each incremental increase of this simulation parameter consistently improves the performance in all test-set partitions, with 15%, 21% 28% relative improvements reported overall for each experimental model. Comparing models 3 and 0, we observe a large reduction in deletion errors which is as high as 52% relative. Breaking down model performance by regions with fully overlapped and non-overlapped speech, we observe that model 3 achieves larger gains on the latter (31% relative WER improvement), while performance gains on the fomer is slightly less pronounced (24.3%). Model 3 still performs significantly worse on fully overlapped segments (50.6% WER) than on non-overlapped ones (16.3%), which reveals that separation in the encoder is far from ideal and can be improved.

Turn counting accuracy – In this section we investigate transcription segmentation quality by comparing the number of turns estimated by the model with the real number of turns in the audio. To do so, we firstly introduce <cot> token into the model vocabulary and target transcriptions as described in Section 3.3. Resulting model is trained using the same audio simulation configuration as model 3, and is denoted as model 4 in Table 2. Its speech recognition performance is slightly worse than the corresponding model without rich transcription capability, which we attribute to the increased deletion rate by 30% relative. We report its turn counting accuracy in Table 3. According to the results, model tends to consistently underestimate the number of turns. Turn counting accuracy (in bold) drops below 50% for utterances with more than 3 turns and it is around 0% for utterances with more than 6 turns (omitted in the table). We attribute it to the fact that the model was never exposed to audio with more than 5 turns. This observation clearly identifies the room for potential improvement of the proposed speech segmentation strategy. Finding a better trade-off between under-segmentation and over-segmentation is of particular importance for downstream application of speaker labeling on top of MT-RNN-T output. One could argue that over-segmentation is preferable here since it is less detrimental and potentially recoverable as part of the speaker labeling pipeline.

Table 3: Confusion matrix for number of turns estimated on LibriCSS by model 4: MT-RNN-T with <cot> token.

Actual	Estimated # turns					
# turns	1	2	3	4	5	6
1	**100.00**	0.00	0.00	0.00	0.00	0.00
2	9.38	**90.34**	0.28	0.00	0.00	0.00
3	1.53	40.46	**58.02**	0.00	0.00	0.00
4	0.00	16.67	48.15	**35.19**	0.00	0.00
5	0.00	6.90	29.31	58.62	**5.17**	0.00
6	0.00	0.00	16.22	62.16	18.92	**2.70**

5. CONCLUSIONS

In this paper we successfully scaled up streaming multi-speaker RNN-T to audio with any number of speakers without compromising streaming ability or changes in the model architecture. The on-the-fly data simulation system stands at the heart of the proposed work. It can simulate speech audio with any number of speakers, both close-talking and far-field, with dynamic multi-turn target arrangement and speech energy variation. On close-talking two-speaker LibriSpeechMix we achieved the new state-of-the-art 8.8% WER among streaming model architectures. We proposed overlap-based target arrangement in a MT-RNN-T model that re-uses the same output channel when there is no speech overlap. We trained this model on data with up to 5 speakers, and benchmarked it against two-speaker MS-RNN-T model on LibriCSS test set. We proposed ORC WER for factored study on word recognition performance to address the challenges in scoring long form output from models with two parallel output channels. In addition, we experimented with joint speech recognition and segmentation, as the first step to rich transcriptions for streaming speaker attributed ASR. Through turn counting analysis of segmentation performance, we demonstrated that MT-RNN-T models with turn segmentation tokens currently tend to underestimate the number of speech segments, which lays the foundation for potential future work.

6. REFERENCES

[1] T. Hain, L. Burget, J. Dines, G. Garau, V. Wan, M. Karafiat, J. Vepa, and M. Lincoln, “The AMI system for the transcription of speech in meetings,” in *Proc. ICASSP 2007*, vol. 4, 2007, pp. IV–357–IV–360.

[2] C. Fox, Y. Liu, E. Zwyssig, and T. Hain, “The Sheffield wargames corpus,” in *Proc. Interspeech 2013*, 2013, pp. 1116–1120.

[3] Y. Liu, C. Fox, M. Hasan, and T. Hain, “The sheffield wargame corpus - day two and day three,” *Interspeech*, Sep 2016.

[4] J. Barker, S. Watanabe, E. Vincent, and J. Trmal, “The fifth ‘CHiME’ speech separation and recognition challenge: Dataset, task and baselines,” *Interspeech*, Sep 2018.

[5] Y. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey, “Single-channel multi-speaker separation using deep clustering,” *Interspeech*, Sep 2016.

[6] T. Menne, I. Sklyar, R. Schlüter, and H. Ney, “Analysis of deep clustering as preprocessing for automatic speech recognition of sparsely overlapping speech,” *Interspeech*, Sep 2019.

[7] S. Settle, J. L. Roux, T. Hori, S. Watanabe, and J. R. Hershey, “End-to-end multi-speaker speech recognition,” in *ICASSP*, 2018, pp. 4819–4823.

[8] 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,” *ICASSP*, May 2020.

[9] T. v. 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,” *Interspeech 2020*, Oct 2020.

[10] D. Yu, M. Kolbaek, Z.-H. Tan, and J. Jensen, “Permutation invariant training of deep models for speaker-independent multi-talker speech separation,” *ICASSP*, Mar 2017.

[11] D. Yu, X. Chang, and Y. Qian, “Recognizing multi-talker speech with permutation invariant training,” *Interspeech*, Aug 2017.

[12] Y. Qian, X. Chang, and D. Yu, “Single-channel multi-talker speech recognition with permutation invariant training,” *Speech Communication*, vol. 104, p. 1–11, Nov 2018.

[13] H. Seki, T. Hori, S. Watanabe, J. Le Roux, and J. R. Hershey, “A purely end-to-end system for multi-speaker speech recognition,” *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018.

[14] X. Chang, Y. Qian, K. Yu, and S. Watanabe, “End-to-end monaural multi-speaker ASR system without pretraining,” *ICASSP*, May 2019.

[15] A. Tripathi, H. Lu, and H. Sak, “End-to-end multi-talker overlapping speech recognition,” in *ICASSP*, 2020, pp. 6129–6133.

[16] N. Kanda, Y. Gaur, X. Wang, Z. Meng, and T. Yoshioka, “Serialized output training for end-to-end overlapped speech recognition,” *Interspeech*, Oct 2020.

[17] N. Kanda, G. Ye, Y. Wu, Y. Gaur, X. Wang, Z. Meng, Z. Chen, and T. Yoshioka, “Large-scale pre-training of end-to-end multi-talker ASR for meeting transcription with single distant microphone,” in *Proc. Interspeech 2021*, 2021, pp. 3430–3434.

[18] 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,” *Interspeech*, Oct 2020.

[19] N. Kanda, X. Chang, Y. Gaur, X. Wang, Z. Meng, Z. Chen, and T. Yoshioka, “Investigation of end-to-end speaker-attributed ASR for continuous multi-talker recordings,” in *Proc. SLT 2021*, 2021, pp. 809–816.

[20] N. Kanda, G. Ye, Y. Gaur, X. Wang, Z. Meng, Z. Chen, and T. Yoshioka, “End-to-end speaker-attributed ASR with transformer,” *Interspeech*, Sep 2021.

[21] N. Kanda, X. Xiao, J. Wu, T. Zhou, Y. Gaur, X. Wang, Z. Meng, Z. Chen, and T. Yoshioka, “A comparative study of modular and joint approaches for speaker-attributed asr on monaural long-form audio,” *ASRU*, 2021.

[22] A. Graves, “Sequence transduction with recurrent neural networks,” *CoRR*, vol. abs/1211.3711, 2012.

[23] I. Sklyar, A. Piunova, and Y. Liu, “Streaming multi-speaker ASR with RNN-T,” in *ICASSP*, 2021, pp. 6903–6907.

[24] 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.

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

[26] T. von Neumann, K. Kinoshita, C. Boeddeker, M. Delcroix, and R. Haeb-Umbach, “Graph-PIT: Generalized permutation invariant training for continuous separation of arbitrary numbers of speakers,” in *Proc. Interspeech 2021*, 2021, pp. 3490–3494.

[27] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: An ASR corpus based on public domain audio books,” in *ICASSP*, 2015, pp. 5206–5210.

[28] T. Kudo and J. Richardson, “SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Nov. 2018, pp. 66–71.

[29] D. S. Park, Y. Zhang, C.-C. Chiu, Y. Chen, B. Li, W. Chan, Q. V. Le, and Y. Wu, “SpecAugment on large scale datasets,” *ICASSP*, May 2020.

[30] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA*, 2015.

[31] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, “SpecAugment: A simple data augmentation method for automatic speech recognition,” *Interspeech*, Sep 2019.

[32] 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 *ICASSP*, 2020, pp. 7284–7288.

[33] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*. USA: Prentice-Hall, Inc., 1993.

[34] X. Huang, A. Acero, H.-W. Hon, and R. Reddy, *Spoken Language Processing: A Guide to Theory, Algorithm, and System Development*, 1st ed. USA: Prentice Hall PTR, 2001.

[35] J. L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” in *arXiv*, 2016.