CONTINUOUS STREAMING MULTI-TALKER ASR WITH DUAL-PATH TRANSDUCERS

Desh Raj[*1], *Liang Lu*[2], *Zhuo Chen*[2], *Yashesh Gaur*[2], *Jinyu Li*[2]

[1]Center for Language and Speech Processing, Johns Hopkins University, USA, [2]Microsoft Corp., USA

ABSTRACT

Streaming recognition of multi-talker conversations has so far been evaluated only for 2-speaker *single-turn* sessions. In this paper, we investigate it for *multi-turn* meetings containing multiple speakers using the Streaming Unmixing and Recognition Transducer (SURT) model, and show that naively extending the single-turn model to this harder setting incurs a performance penalty. As a solution, we propose the dual-path (DP) modeling strategy first used for time-domain speech separation. We experiment with LSTM and Transformer based DP models, and show that they improve word error rate (WER) performance while yielding faster convergence. We also explore training strategies such as chunk width randomization and curriculum learning for these models, and demonstrate their importance through ablation studies. Finally, we evaluate our models on the LibriCSS meeting data, where they perform competitively with offline separation-based methods.

***Index Terms*—** Multi-talker ASR, long-form meeting transcription, dual-path RNN, transducer

1. INTRODUCTION

With advancements in clean, single-speaker transcription [1, 2], researchers are focusing on harder speech recognition settings with interference in the form of noise, reverberation, or overlapping speakers [3, 4, 5]. Of these, the latter is particularly irksome, while also being prevalent in meetings and natural conversations [6, 7, 8]. The conventional approach to handle overlapping speech is through a cascade of separation and recognition modules, which may be sub-optimal since the modules are independently optimized. It also requires considerable engineering efforts to maintain the pipeline.

An alternative solution to this problem is through the application of end-to-end models that are trained with text-based supervision [9, 10, 11, 12, 13, 14, 15, 16]. Several model paradigms have been explored, including attention-based encoder-decoders [17, 18, 19], and recurrent neural network transducers (RNN-T) [20, 21], with most of the models functioning in an offline mode. For the streaming case, Lu et al. recently proposed Streaming Unmixing and Recognition Transducer (SURT) [22] which simultaneously transcribes overlapping speech into two channels through

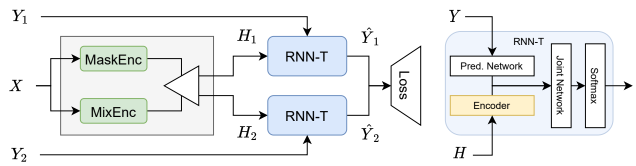


Fig. 1: An overview of the RNN-T based SURT model for the 2-speaker overlapping case.

"unmixing" and "transcription" modules that are jointly optimized using an RNN-T based heuristic error assignment training (HEAT) loss. We will review SURT in § 2.

SURT was originally evaluated on 2-speaker single-turn sessions and showed promising results. However, real-life meetings often contain multiple speakers and several turns of conversation, and it is not immediately clear whether the "vanilla" SURT model would transfer well to these harder settings. In this paper, we show that the train-test mismatch does, in fact, incur performance costs that cannot be easily recovered using multi-turn training. We then investigate dual-path (DP) LSTM and Transformer models, which have been successfully applied in speech separation [23, 24], and show that they are better at modeling the longer sequences that occur in multi-turn meetings. Our contributions include training strategies for DP models, such as curriculum learning and chunk width randomization, and a theoretical analysis of DP Transformers. We evaluate our models on several tiers of simulated multi-talker mixtures and on the LibriCSS meeting corpus [25], where it performs competitively with offline separation-based approaches.

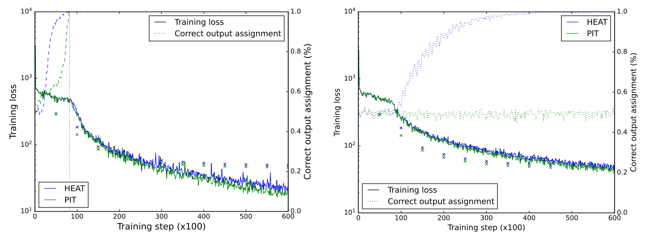
2. SURT: REVIEW

Fig. 1 shows an overview of the RNN-T based SURT model. Given input features X obtained from an overlapping speech session, a mask-based unmixing module extracts speaker-dependent representations H_1 and H_2 as

$$\begin{aligned} &H_1 = M * \bar{X}, \quad H_2 = (\mathbb{1} - M) * \bar{X}, \\ &M = \sigma(\texttt{MaskEnc}(X)) \text{ and } \bar{X} = \texttt{MixEnc}(X). \end{aligned} \tag{1}$$

Here, σ denotes the Sigmoid function, $*$ is the Hadamard product, and $\mathbb{1}$ has the same shape as M. Both the MaskEnc and MixEnc use a 4-layer 2D convolutional architecture. H_1 and H_2 are fed into the RNN-T module, and produce hypotheses $\hat{Y}_1$ and $\hat{Y}_2$ under the HEAT assumption (i.e., Y_1 starts be-

[*]Work done during internship at Microsoft.



(a) Delay = 2.0 s **(b)** Delay = 0.0 s

Fig. 2: Training dynamics for HEAT versus PIT based loss for different utterance delays: (a) 2.0 s, and (b) 0.0 s.

fore Y_2). The final loss is given as

$$\mathcal{L}_{\text{heat}}(X, Y_1, Y_2) = \mathcal{L}_{\text{rnnt}}(Y_1, H_1) + \mathcal{L}_{\text{rnnt}}(Y_2, H_2), \quad (2)$$

where $\mathcal{L}_{\text{rnnt}}$ is the standard RNN-T loss. Although Lu et al. only evaluated HEAT on 2-speaker single-turn sessions, it is straightforward to extend HEAT to long-form sessions.

2.1. HEAT vs. PIT

The output label permutation problem in overlapped ASR is often addressed using utterance-level permutation invariant training (PIT), which computes the optimal permutation of reference and hypothesis sequences as the training loss. For the SURT problem formulated above, it is given as

$$\mathcal{L}_{\text{pit}}(X, Y_1, Y_2) = \min\big(\mathcal{L}_{\text{rnnt}}(Y_1, H_1) + \mathcal{L}_{\text{rnnt}}(Y_2, H_2), \\ \mathcal{L}_{\text{rnnt}}(Y_1, H_2) + \mathcal{L}_{\text{rnnt}}(Y_2, H_1)\big). \quad (3)$$

In real use cases, we usually have information (such as pitch or gender) which can disambiguate the label sequences. For long-form meetings with partially overlapped utterances, utterance start time can be a good heuristic for output matching, and has been shown to outperform PIT [22]. To investigate this phenomenon further, we set up simple experiments on 2-utterance mixtures generated from LibriSpeech `train-clean` set. We prepared two kinds of mixtures with utterance delays of 2.0 and 0.0 seconds, respectively, and trained the vanilla SURT model using both HEAT and PIT losses (equations 2 and 3, respectively). In Fig. 2, we show the training dynamics for both the experiments and also plot the % correct output assignment. This quantity represents how often the model assigns Y_1 to output channel 1.

As expected, for the case of mixtures with 2.0s delay, HEAT quickly learned the output assignment order. In fact, even when training with PIT, the same heuristic was learned (albeit slower), and both models started to converge only after this point was reached (denoted by the vertical line in Fig. 2(a). Thereafter, using PIT is wasteful, especially in our case of the expensive RNN-T loss computation. In the absence of utterance delay (Fig. 2(b)), PIT produced a random output assignment. Surprisingly, HEAT still learned the correct assignment, but on decoding with the trained model, we found that it learned a degenerate solution where both output channels produce the exact same hypothesis.

Table 1: Summary of evaluation sets used in this work.

Name	**Description**	**# spk.**	**# utt.**	`dev`	`test`
Tier-1	2-speaker single-turn	2	2	1355	1310
Tier-2	2-speaker multi-turn	2	2-4	892	885
Tier-3	Multi-speaker multi-turn	2-4	2-12	462	450

Besides HEAT and PIT being empirically equivalent for non-zero delays, it is infeasible to compute all permutations when training on longer sessions, since the complexity grows as $\mathcal{O}(N!)$, with N being the number of utterances. Even if we use linear sum assignment (which has a complexity of $\mathcal{O}(N^3)$ [26]), it still requires computing the loss between all N^2 pairs of references and hypotheses, which is computationally prohibitive when using RNN-T since it implicitly sums over all possible alignments. For these reasons, we use HEAT for all experiments in the remainder of this paper.

3. MULTI-TURN EVALUATION

Our objective is to investigate SURT for long-form multi-talker sessions. For this, we first prepared 3 tiers of evaluation data with increasing difficulty, all of which are obtained by mixing LibriSpeech test set utterances. Tier-1 (T1) contains single-turn sessions similar to the original SURT evaluation data [22]. Tier-2 (T2) extends this to sessions containing up to 4 utterances, while retaining 2 speakers. Finally, Tier-3 (T3) contains multi-turn sessions with up to 4 speakers. All the tiers were generated to contain an overlap ratio between 0% and 40% per session. Table 1 summarizes the statistics for these evaluation sets. For training, we prepared 2-speaker single and multi-turn sessions which are comparable to the T1 and T2 evaluation sets, respectively. It is infeasible to train on longer sessions (like T3) because of memory constraints. Throughout this paper, we evaluated our models using the word error rate (WER) metric computed between the best permutation of reference utterances and the output channels[1]. For our preliminary experiments, we considered the vanilla SURT model which uses a 6-layer 1024-dim LSTM network in the RNN-T encoder (see yellow box in Fig. 1). The prediction network is a 2-layer 1024-dim LSTM model. The model performance when trained on single and multi-turn sessions are shown below:

Train \Eval	**Tier-1**	**Tier-2**	**Tier-3**
Single-turn	11.1	17.6	24.9
Multi-turn	13.6	15.9	20.9

Can the vanilla SURT model trained on single-turn sessions generalize to multi-turn evaluation? They cannot. WER degrades significantly as tier complexity increases (row 1), presumably because the model had never seen any instances with multiple utterances in an output channel.

[1]Since the utterances are ordered by start time, this process elicits 2^N permutations, which makes evaluation infeasible for N greater than 12.

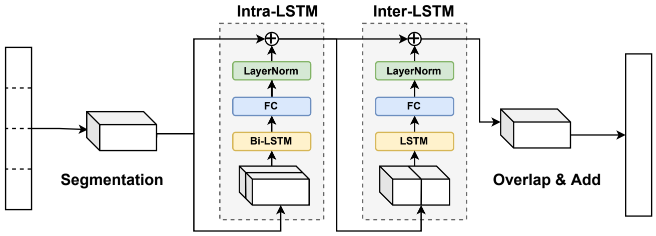


Fig. 3: Overview of the streaming dual path LSTM model. The intra-chunk LSTM is bidirectional, whereas the inter-chunk LSTM is unidirectional.

Can we generalize to multi-turn sessions by simply training on matching data? Although the WER on T2 and T3 improved, such training degraded T1 results (row 2), suggesting that it is challenging to make standard LSTM models generalize to diverse session lengths. An alternate model may be required which is better suited to this problem.

4. STREAMING DUAL-PATH TRANSDUCERS

Single channel time-domain speech separation, which requires modeling extremely long sequences, has benefited from the application of dual-path (DP) networks such as DP-RNN [23] and DP-Transformers [24]. Here, we investigate streaming versions of these models as encoders in the SURT RNN-T module.

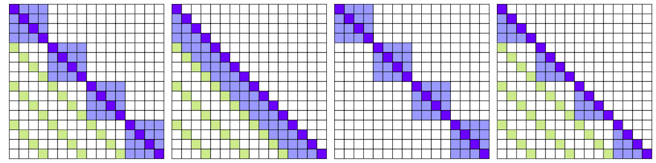
4.1. Dual-path LSTM

DP-LSTM consists of an *intra* and an *inter* LSTM per layer [23], as shown in Fig. 3. Input sequences are segmented into (overlapping) chunks and first fed into the bidirectional intra-LSTM, which processes each chunk independently. The output is then passed into the inter-LSTM which is unidirectional and performs strided processing over chunks. By choosing the chunk width to be approximately square root of the sequence length l, we can ensure that both the LSTMs get similar length inputs. Since the intra-LSTM is bidirectional, a latency equal to the chunk width is introduced in this model.

4.2. Dual-path Transformer

On the surface, DP-Transformer is similar to the DP-LSTM model, with the LSTM blocks replaced with self-attention blocks [24]. The intra-Transformer uses a full attention matrix, while the inter-Transformer is constrained to use causal attention. Consequently, the overall attention pattern for the sequence can be visualized as shown in Fig. 4(a). Similar to the DP-LSTM, we can ensure $\mathcal{O}(l)$ attention computation for each chunk by choosing $\mathcal{O}(\sqrt{l})$ chunk sizes. Consequently, the DP-Transformer has a complexity of $\mathcal{O}(l\sqrt{l})$, as opposed to the quadratic complexity of the vanilla Transformer. Sparse transformers with similar attention patterns have previously been proposed for long sequences (Fig.4).

Despite this reduced complexity, the DP-Transformer (in non-streaming mode) is a universal function approximator.



(a) Dual-path **(b)** Strided [27] **(c)** Block [28] **(d)** Axial [29]

Fig. 4: Comparison of self-attention patterns in different $\mathcal{O}(l\sqrt{l})$ transformer architectures.

Table 2: WER results for SURT models with regular and dual-path encoders, trained on multi-turn data with curriculum learning.

Encoder	Size	Tier-1		Tier-2		Tier-3	
		dev	test	dev	test	dev	test
LSTM	75.6 M	13.6	13.8	15.9	17.1	20.9	21.0
DP-LSTM	65.4 M	**11.1**	11.7	13.7	14.7	19.9	20.2
+ CWR		**11.1**	**11.4**	**13.0**	**14.1**	19.6	19.6
DP-Transformer	42.9 M	11.5	12.4	13.7	15.1	19.1	20.4
+ CWR		**11.1**	12.2	13.5	14.5	**17.9**	**18.6**

To see this, consider the following properties for the DP-Transformer. First, every token in the sequence attends to itself. Second, the directed graph $\mathcal{D}$ corresponding to the attention pattern contains a Hamiltonian path through all tokens. Third, any token can directly/indirectly access all other tokens after exactly 2 layers (in non-streaming mode). Therefore, the model satisfies all necessary conditions for universal approximability of sparse transformers [30].

4.3. Chunk width randomization

Dual-path models trained with a fixed chunk width may not be suitable for evaluation on diverse sequence lengths due to mismatch in train-test input size for the inter block. We propose training with chunk width randomization (CWR), wherein we vary the CW between a minimum and maximum value for each mini-batch. CWR increases the train time diversity in sequence length for both the intra and inter blocks and makes the model robust to such variations at test time.

5. EXPERIMENTAL SETUP

The architecture of the vanilla SURT model is detailed in §3. For the DP-based models, we replaced the RNN-T encoder with the corresponding dual path modules: a 6-layer DP-LSTM with 512-dim intra and inter blocks, and a 12-layer DP-Transformer consisting of 256-dim self-attention split into 8 heads with a 1024-dim feedforward layer. All models were trained to convergence using a learning rate schedule consisting of 10k steps of warmup to 0.0003 and a linear decay thereafter. We used AdamW as the optimizer and clipped gradients to a norm of 5. The models were trained on 16 V100 GPUs using FP16 precision. We used decoding beams of size 4 and 8 for the `dev` and `test` sets, respectively, in all experiments. For the fixed chunk dual-path models, a chunk width of 30 was used for training and decoding. For models with CWR, we trained with chunk widths between 15 and

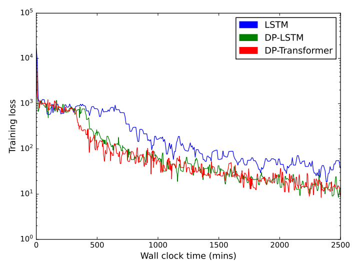


Fig. 5: Training curves for the dual-path models compared with the regular LSTM model, plotted w.r.t. wall clock time (mins).

45 and decoded with chunks of width 35. These values were tuned on the `dev` set. For the LibriCSS experiments, since the data contains far-field recordings [25], we additionally trained using simulated noise and reverberations [31]. For these evaluations, we decoded on the approx. 1-minute long segments provided with the corpus (cf. note 1).

6. RESULTS & DISCUSSION

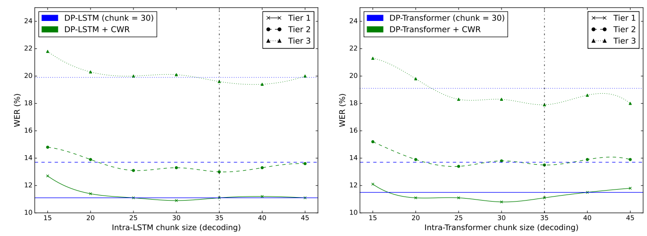
Multi-turn evaluation. Table 2 shows the results for our DP models compared to the LSTM-based SURT. Both the DP-LSTM and DP-Transformer provide WER improvements across all evaluation tiers without degrading performance on single-turn sessions (§3). CWR training helped both models, with substantial gains for DP-Transformer on T3 evaluation. DP-Transformer are competitive with DP-LSTM, and outperform them on T3, with only two-thirds the model size.

Training dynamics and curriculum learning. Fig. 5 shows the training curves for the dual-path models compared with the LSTM-based SURT model. Both the DP-LSTM and DP-Transformer converged faster and to a better minimum, despite smaller model sizes. We also found it important to use a curriculum learning strategy for all the models, where we train them first on single-turn sessions, and then introduce multi-turn sessions after 20k training steps. Table 3 compares performances on the `dev` set for models trained with and without a curriculum. Since the multi-turn training data also contains single-turn sessions, and the benefits are seen across all tiers, these improvements cannot be attributed solely to variation in training data.

Accuracy vs. latency. Besides improving robustness to session lengths, CWR also enables models to be deployed with lower latency at the cost of a small WER degradation. Fig. 6 shows this trade-off between decoding latency (determined by chunk width) and WER performance. WER tends to degrade slightly for low latency decoding, and the best performance is

Table 3: Effect of curriculum learning on `dev` set. We compare models trained only on multi-turn sessions versus those initialized from 20k steps of single-turn training.

Model	Tier-1		Tier-2		Tier-3	
	None	*20k*	*None*	*20k*	*None*	*20k*
LSTM	15.8	13.6 ($\downarrow$14.1%)	17.8	15.9 ($\downarrow$10.7%)	21.8	20.9 ($\downarrow$4.1%)
DP-LSTM	11.4	11.1 ($\downarrow$2.6%)	13.6	13.0 ($\downarrow$4.4%)	18.6	19.6 ($\uparrow$5.4%)
DP-Transformer	12.5	11.1 ($\downarrow$11.2%)	13.7	13.5 ($\downarrow$1.5%)	18.2	17.9 ($\downarrow$1.6%)



(a) DP-LSTM **(b)** DP-Transformer

Fig. 6: Accuracy vs. latency trade-off for dual-path models trained with chunk width randomization (CWR), evaluated on `dev` set.

Table 4: WER results for single-channel LibriCSS. 0L and 0S denote 0% overlap with long and short silences, respectively.

Model		Overlap ratio in %					
		0L	**0S**	**10**	**20**	**30**	**40**
BLSTM CSS + Hybrid ASR [25]		16.3	17.6	20.9	26.1	32.6	36.1
Conformer CSS + E2E ASR [32]		6.1	6.9	9.1	12.5	16.7	19.3
SURT w/ DP-LSTM	*replayed*	9.8	19.1	20.6	20.4	23.9	26.8
SURT w/ DP-LSTM	*clean*	6.6	21.6	21.7	20.6	25.4	28.4
SURT w/ DP-Transformer	*replayed*	9.3	21.1	21.2	25.9	28.2	31.7
SURT w/ DP-Transformer	*clean*	6.9	18.9	19.6	21.9	23.9	28.7

obtained for chunk sizes of 35.

Evaluation on LibriCSS. Table 4 compares our models against *offline* modular systems composed of continuous speech separation (CSS) and ASR modules. The "conformer CSS + E2E ASR" model [32] contains 197M parameters (59M for CSS, and 138M for ASR [33]), and uses language model fusion and rescoring. On high overlap conditions, our models are competitive with these state-of-the-art offline modular systems, with most errors arising from *leakage* in single-speaker regions, i.e., when both channels transcribed an utterance even when no overlap occurred. For the low overlap cases, *omissions* were another major error source, where some utterances were completely missed by both channels. DP-Transformer does not outperform DP-LSTM in this case possibly due to overfitting to the simulated noise and reverberation, as evident from their performance in the *clean* (digitally mixed clean audio) setting.

7. CONCLUSION

We investigated SURT for continuous streaming multi-talker ASR, and demonstrated the effectiveness of dual-path LSTMs and Transformers for generalization to diverse session lengths. Training strategies such as curriculum learning and chunk width randomization provided WER improvements and enabled the model to be deployed with different latencies. We also showed that DP-Transformers are universal function approximators. Our models demonstrated encouraging results across different overlap levels on the LibriCSS dataset while being smaller, faster, and simpler than modular systems. In future work, we will explore segmentation in the SURT output channels, which will naturally enable speaker-attributed ASR.

8. REFERENCES

[1] D. Amodei et al., “Deep speech 2 : End-to-end speech recognition in english and mandarin,” in *Proceedings of The 33rd International Conference on Machine Learning*, 2016.

[2] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, “Toward human parity in conversational speech recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2017.

[3] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, “The third ‘CHiME’ speech separation and recognition challenge: Dataset, task and baselines,” in *IEEE ASRU*, 2015.

[4] K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, A. Sehr, W. Kellermann, and R. Maas, “The REVERB challenge: A common evaluation framework for dereverberation and recognition of reverberant speech,” in *IEEE WASPAA*, 2013.

[5] S. Watanabe, M. Mandel, J. Barker, and E. Vincent, “CHiME-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings,” *ArXiv*, vol. abs/2004.09249, 2020.

[6] J. Carletta et al., “The AMI meeting corpus: A pre-announcement,” in *MLMI*, 2005.

[7] E. Shriberg, A. Stolcke, and D. Baron, “Observations on overlap: findings and implications for automatic processing of multi-party conversation,” in *INTERSPEECH*, 2001.

[8] T. Yoshioka, D. Dimitriadis, A. Stolcke, W. Hinthorn, Z. Chen, M. Zeng, and X. Huang, “Meeting transcription using asynchronous distant microphones,” in *INTERSPEECH*, 2019.

[9] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *ICML*, 2006.

[10] A. Graves, “Sequence transduction with recurrent neural networks,” in *ICML Representation Learning Workshop*, 2012.

[11] L. Lu, X. Zhang, and S. Renals, “On training the recurrent neural network encoder-decoder for large vocabulary end-to-end speech recognition,” in *IEEE ICASSP*, 2016.

[12] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, “Attention-based models for speech recognition,” in *Advances in Neural Information Processing Systems*, 2015.

[13] C. Chiu, T. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J. Weiss, K. Rao, K. Gonina, N. Jaitly, B. Li, J. Chorowski, and M. Bacchiani, “State-of-the-art speech recognition with sequence-to-sequence models,” in *IEEE ICASSP*, 2018.

[14] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang, et al., “Streaming end-to-end speech recognition for mobile devices,” in *IEEE ICASSP*, 2019.

[15] J. Li, R. Zhao, H. Hu, and Y. Gong, “Improving RNN transducer modeling for end-to-end speech recognition,” in *IEEE ASRU*, 2019.

[16] J. Li, “Recent advances in end-to-end automatic speech recognition,” *arXiv preprint arXiv:2111.01690*, 2021.

[17] S. Settle, J. Le Roux, T. Hori, S. Watanabe, and J. Hershey, “End-to-end multi-speaker speech recognition,” in *IEEE ICASSP*, 2018.

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

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

[20] A. Tripathi, H. Lu, and H. Sak, “End-to-end multi-talker overlapping speech recognition,” in *IEEE ICASSP*, 2020.

[21] I. Sklyar, A. Piunova, and Y. Liu, “Streaming multi-speaker ASR with RNN-T,” in *IEEE ICASSP*, 2021.

[22] L. Lu, N. Kanda, J. Li, and Y. Gong, “Streaming end-to-end multi-talker speech recognition,” *IEEE Signal Processing Letters*, 2021.

[23] Y. Luo, Z. Chen, and T. Yoshioka, “Dual-path RNN: Efficient long sequence modeling for time-domain single-channel speech separation,” in *IEEE ICASSP*, 2020.

[24] J-J. Chen, Q. Mao, and D. Liu, “Dual-path transformer network: Direct context-aware modeling for end-to-end monaural speech separation,” in *INTERSPEECH*, 2020.

[25] Z. Chen, T. Yoshioka, L. Lu, T. Zhou, Z. Meng, Y. Luo, J. Wu, and J. Li, “Continuous speech separation: Dataset and analysis,” in *IEEE ICASSP*, 2020.

[26] H. Kuhn, “The hungarian method for the assignment problem,” *Naval Research Logistics Quarterly*, vol. 2, pp. 83–97, 1955.

[27] R. Child, S. Gray, A. Radford, and I. Sutskever, “Generating long sequences with sparse transformers,” *ArXiv*, vol. abs/1904.10509, 2019.

[28] J. Qiu, H. Ma, O. Levy, W-T. Yih, S. Wang, and J. Tang, “Blockwise self-attention for long document understanding,” in *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2020.

[29] J. Ho, N. Kalchbrenner, D. Weissenborn, and T. Salimans, “Axial attention in multidimensional transformers,” *ArXiv*, 2019.

[30] C. Yun, Y-W. Chang, S. Bhojanapalli, A. S. Rawat, S. Reddi, and S. Kumar, “O(n) connections are expressive enough: Universal approximability of sparse transformers,” in *Advances in Neural Information Processing Systems*, 2020.

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

[32] S. Chen, Y. Wu, Z. Chen, J. Li, C. Wang, S. Liu, and M. Zhou, “Continuous speech separation with conformer,” in *IEEE ICASSP*, 2021.

[33] C. Wang, Y. Wu, Y. Du, J. Li, S. Liu, L. Lu, S. Ren, G. Ye, S. Zhao, and M. Zhou, “Semantic mask for transformer based end-to-end speech recognition,” in *INTERSPEECH*, 2020.