STREAMING MULTI-SPEAKER ASR WITH RNN-T

Ilya Sklyar
Anna Piunova
Yulan Liu

06.2021

{ilsklyar, piunova, lyulan}@amazon.com

IEEE ICASSP 2021

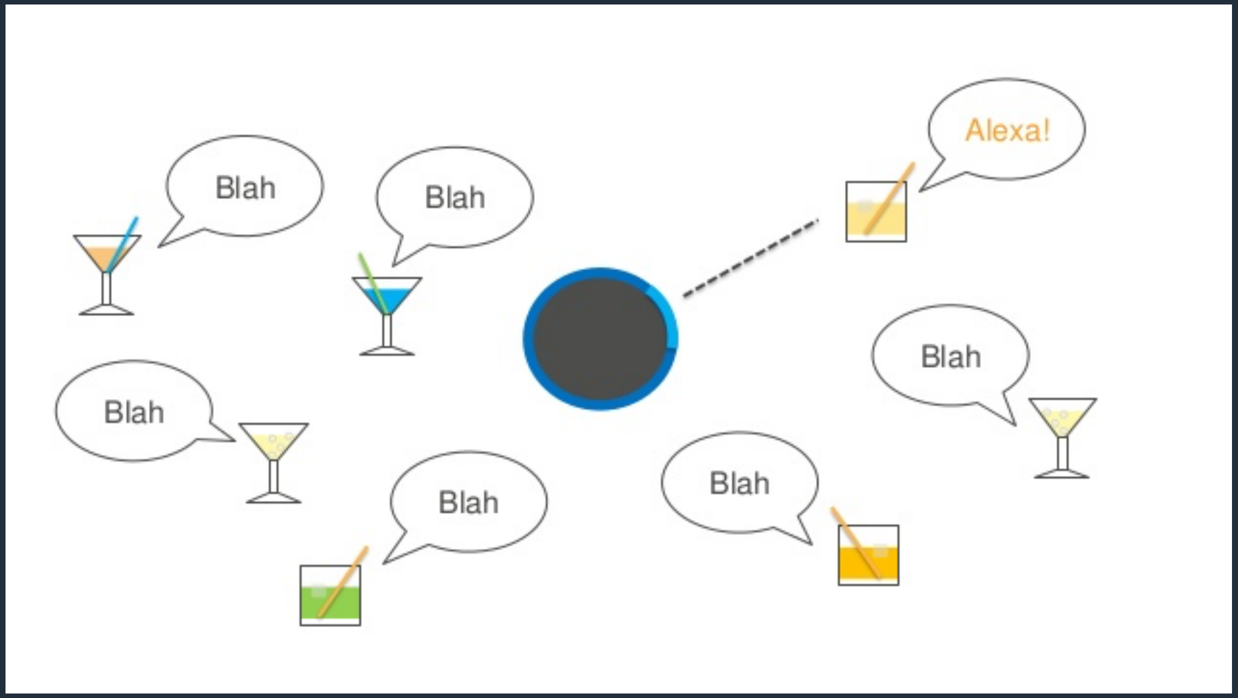
amazon alexa

Motivation

- **Cocktail party problem**: separating and recognizing all speakers in the audio stream

- Current approach: ASR system is trained to recognize single-speaker device-directed speech and ignore all interference

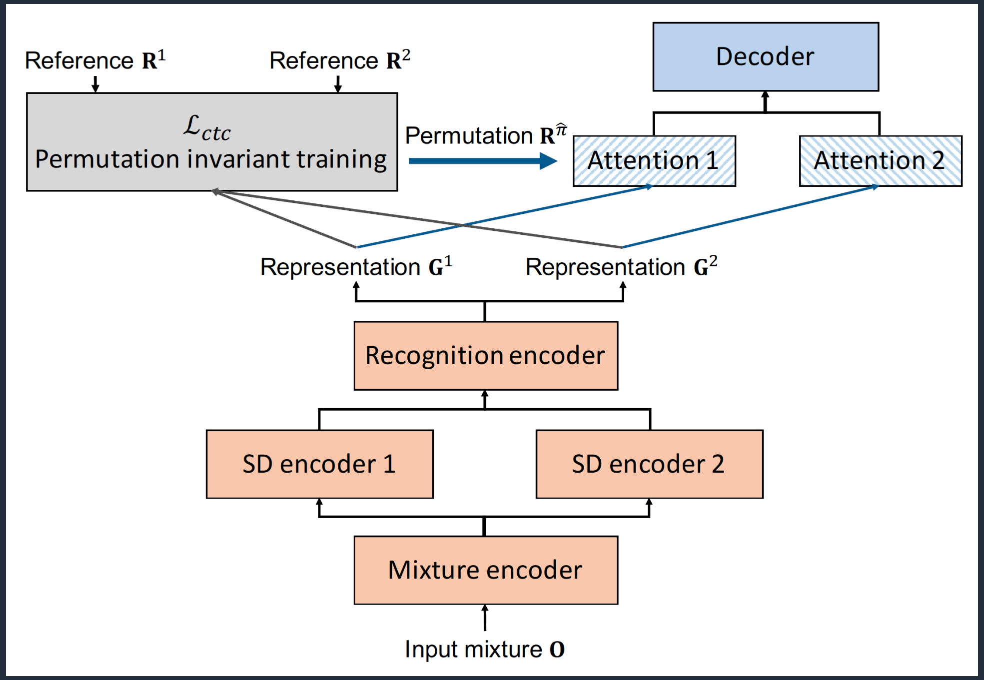
- Ideal conversational ASR: wakeword-free multi-party interactions
  - Overlapping speech processing capability
  - Low-latency restriction for streaming
  - Only one channel can be available



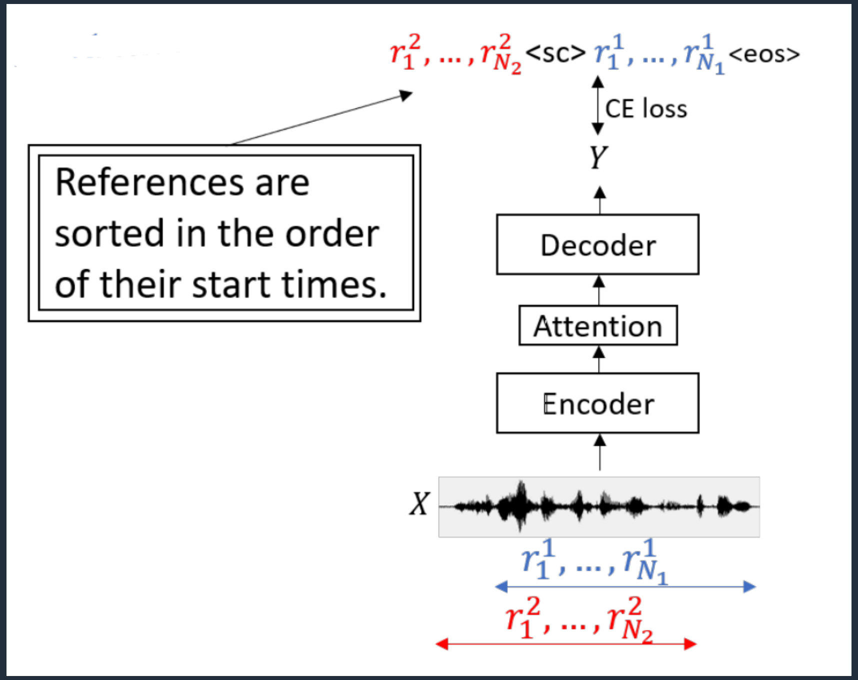
amazon alexa

Prior work on single-channel multi-speaker ASR

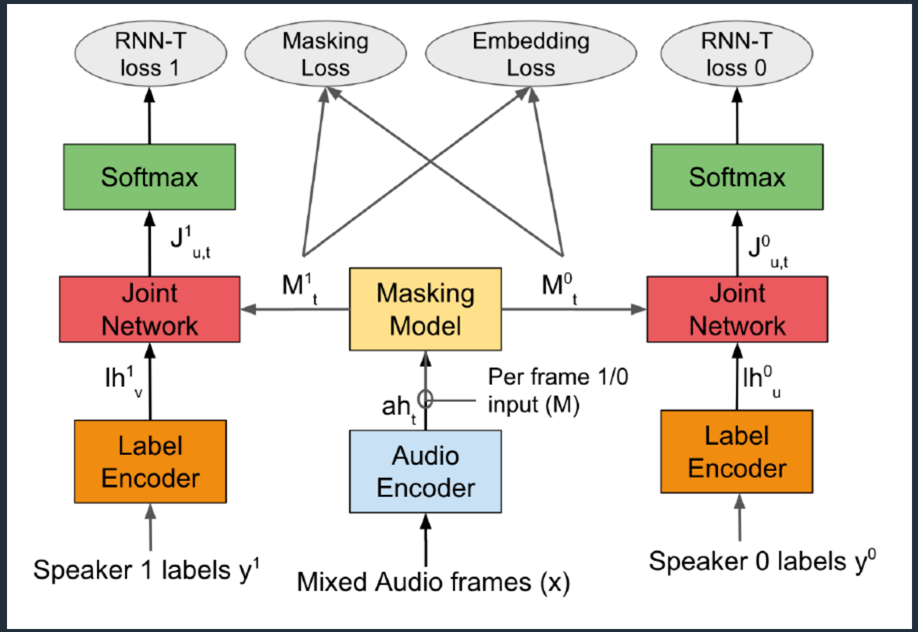
- With source separation objective [1-4]
  - Deep clustering [5]
  - TaSNet [6]
- With single ASR objective
  - Permutation Invariant Training (PIT) [7-11]
  - Serialized Output Training (SOT) [12, 13]



PIT-AED [11]



SOT-AED [12]

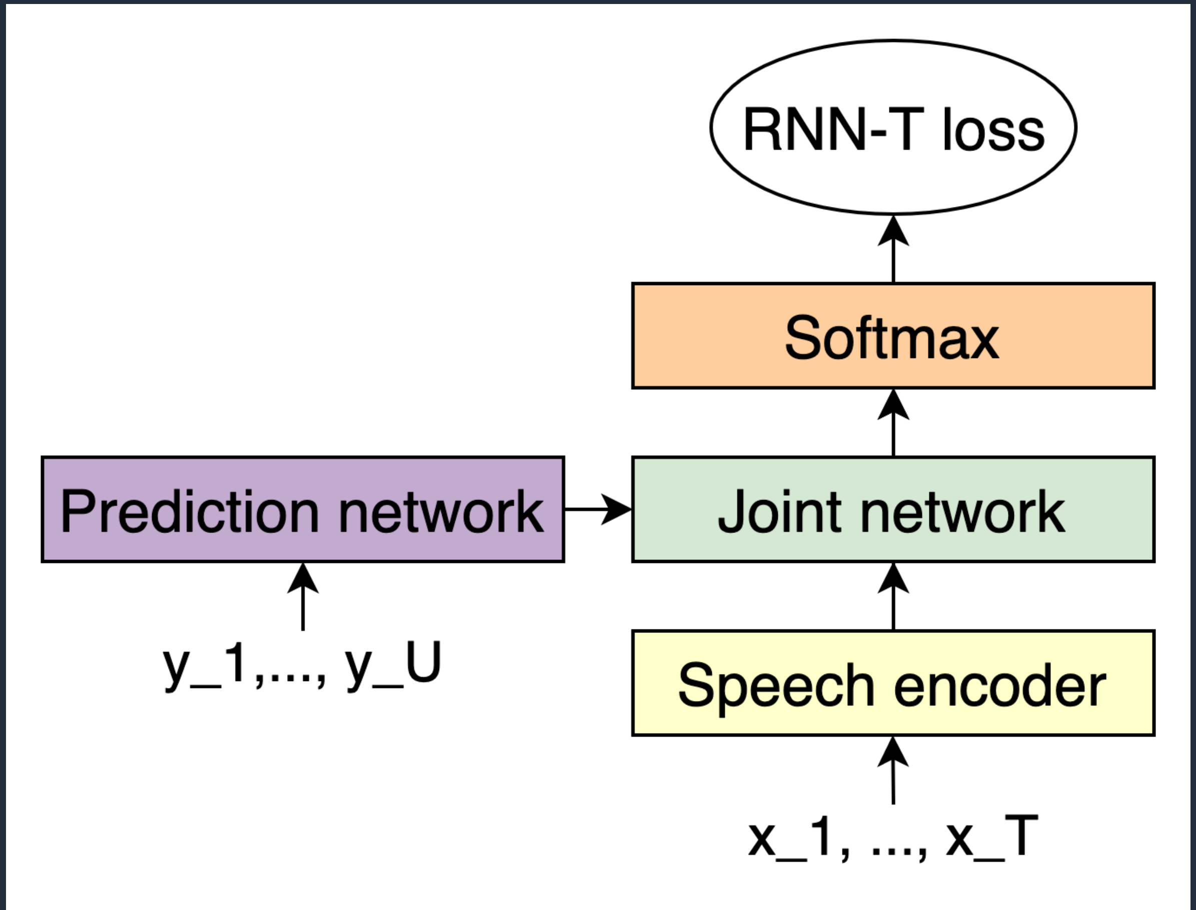


Multi-talker RNN-T [18]

amazon alexa

Main contributions

- First attempt to build a *streaming* multi-speaker ASR system
- Based on Recurrent neural network transducer (RNN-T) [14]
- A study of two different training approaches
  - Deterministic assignment training (DAT)
  - Permutation Invariant Training (PIT)
- On-par results with a non-streaming SOT model on partially overlapping speech of 2 speakers



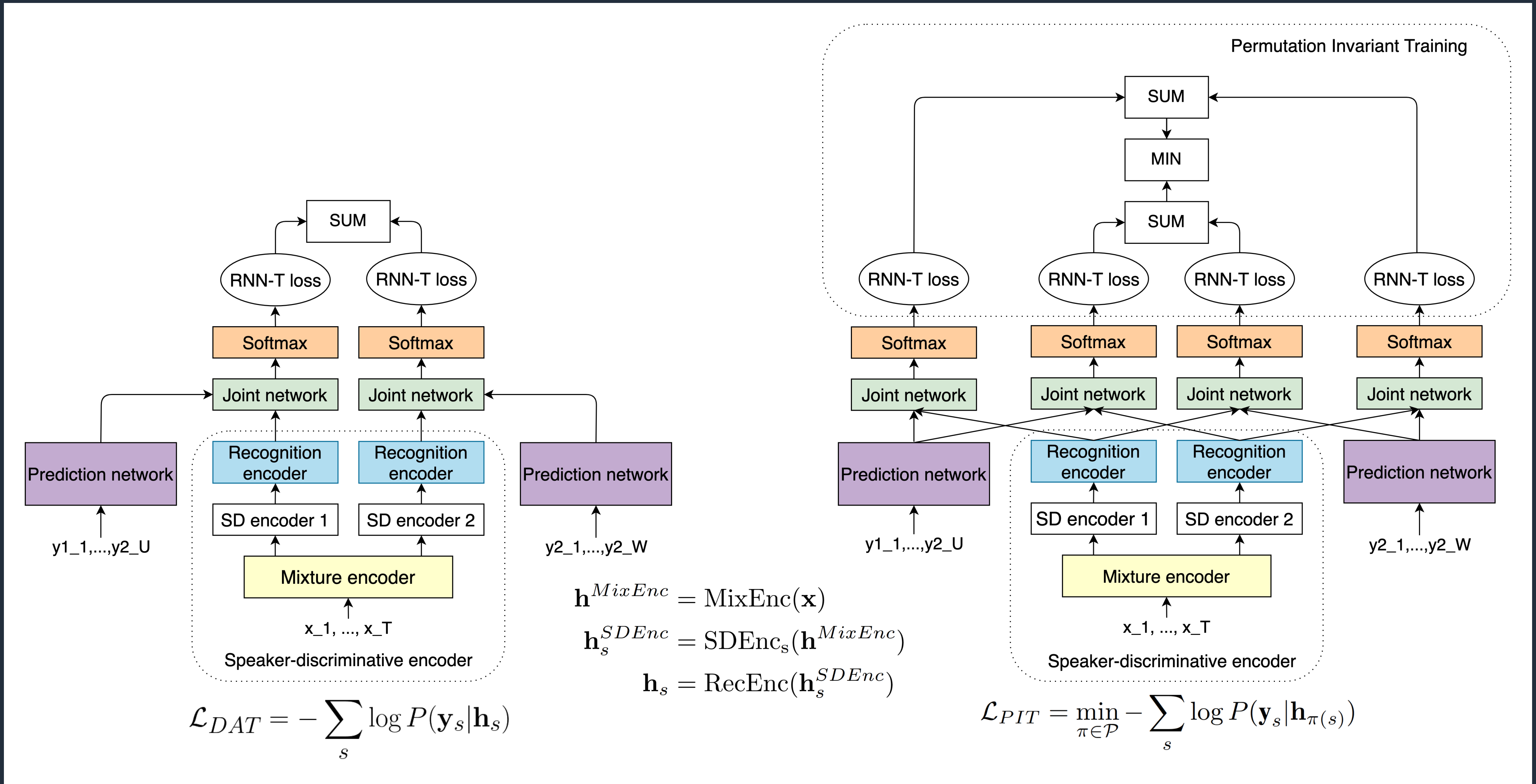
Single-speaker RNN-T

amazon alexa

Multi-speaker RNN-T (MS-RNNT)

DAT-MS-RNN-T

PIT-MS-RNN-T



$$\mathbf{h}^{MixEnc} = \mathrm{MixEnc}(\mathbf{x})$$

$$\mathbf{h}_s^{SDEnc} = \mathrm{SDEnc_s}(\mathbf{h}^{MixEnc})$$

$$\mathbf{h}_s = \mathrm{RecEnc}(\mathbf{h}_s^{SDEnc})$$

$$\mathcal{L}_{DAT} = -\sum_s \log P(\mathbf{y}_s|\mathbf{h}_s)$$

$$\mathcal{L}_{PIT} = \min_{\pi \in \mathcal{P}} -\sum_s \log P(\mathbf{y}_s|\mathbf{h}_{\pi(s)})$$

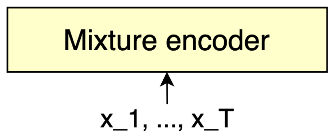
DAT: Deterministic assignment training

PIT: Permutation invariant training

amazon alexa

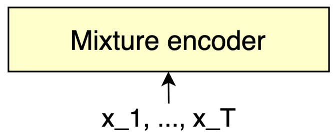
Multi-speaker RNN-T (MS-RNNT)

DAT-MS-RNN-T



$$\mathbf{h}^{MixEnc} = \mathrm{MixEnc}(\mathbf{x})$$

PIT-MS-RNN-T

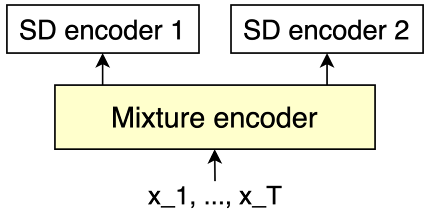


DAT: Deterministic assignment training

PIT: Permutation invariant training

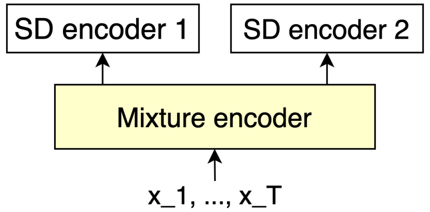
Multi-speaker RNN-T (MS-RNNT)

DAT-MS-RNN-T



$$\mathbf{h}^{MixEnc} = \mathrm{MixEnc}(\mathbf{x})$$
$$\mathbf{h}_s^{SDEnc} = \mathrm{SDEnc_s}(\mathbf{h}^{MixEnc})$$

PIT-MS-RNN-T



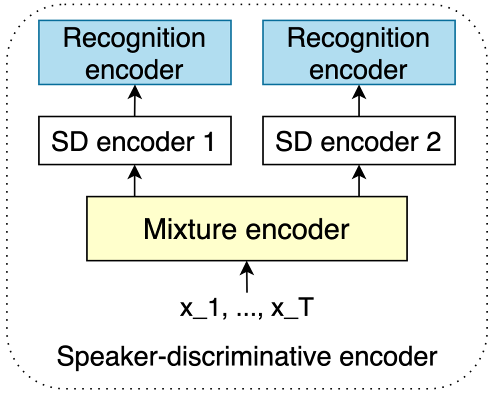
DAT: Deterministic assignment training

PIT: Permutation invariant training

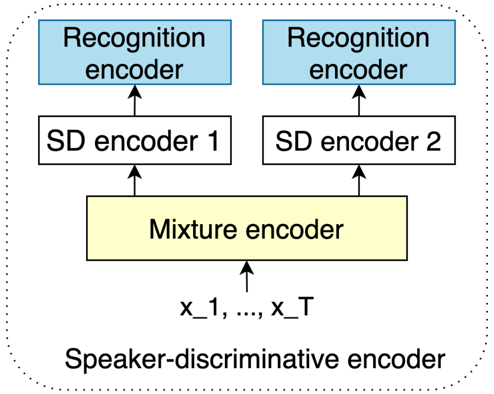
Multi-speaker RNN-T (MS-RNNT)

DAT-MS-RNN-T

PIT-MS-RNN-T



$$
\begin{aligned}
\mathbf{h}^{MixEnc} &= \mathrm{MixEnc}(\mathbf{x}) \\
\mathbf{h}_s^{SDEnc} &= \mathrm{SDEnc_s}(\mathbf{h}^{MixEnc}) \\
\mathbf{h}_s &= \mathrm{RecEnc}(\mathbf{h}_s^{SDEnc})
\end{aligned}
$$



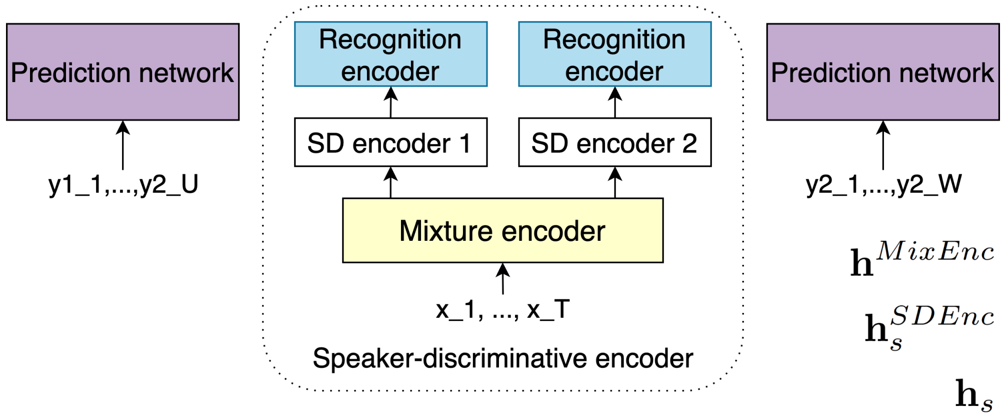
DAT: Deterministic assignment training

PIT: Permutation invariant training

amazon alexa

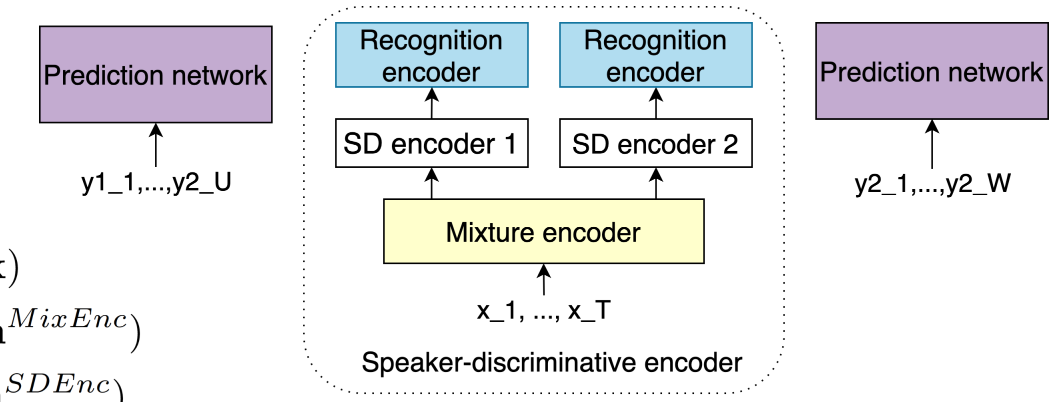
Multi-speaker RNN-T (MS-RNNT)

DAT-MS-RNN-T



$$\mathbf{h}^{MixEnc} = \mathrm{MixEnc}(\mathbf{x})$$
$$\mathbf{h}_s^{SDEnc} = \mathrm{SDEnc}_s(\mathbf{h}^{MixEnc})$$
$$\mathbf{h}_s = \mathrm{RecEnc}(\mathbf{h}_s^{SDEnc})$$

PIT-MS-RNN-T



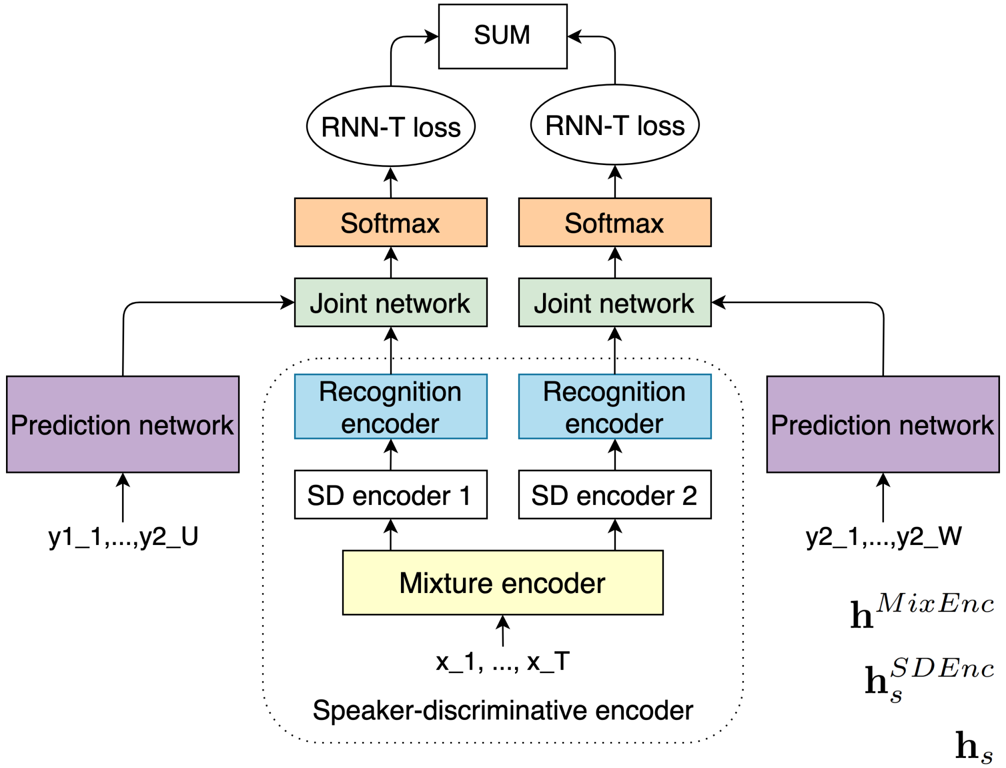
DAT: Deterministic assignment training

PIT: Permutation invariant training

amazon alexa

Multi-speaker RNN-T (MS-RNNT)

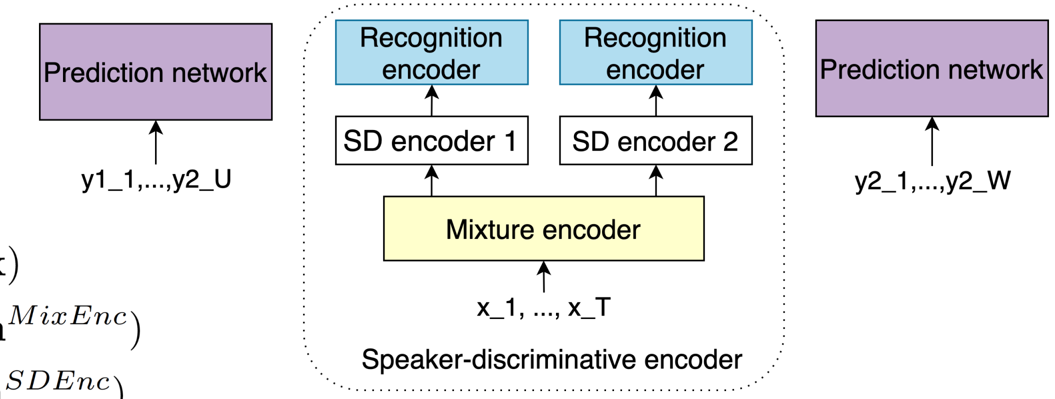
DAT-MS-RNN-T



$$\mathcal{L}_{DAT} = -\sum_{s} \log P(\mathbf{y}_s|\mathbf{h}_s)$$

$$\mathbf{h}^{MixEnc} = \mathrm{MixEnc}(\mathbf{x})$$
$$\mathbf{h}_s^{SDEnc} = \mathrm{SDEnc}_s(\mathbf{h}^{MixEnc})$$
$$\mathbf{h}_s = \mathrm{RecEnc}(\mathbf{h}_s^{SDEnc})$$

PIT-MS-RNN-T



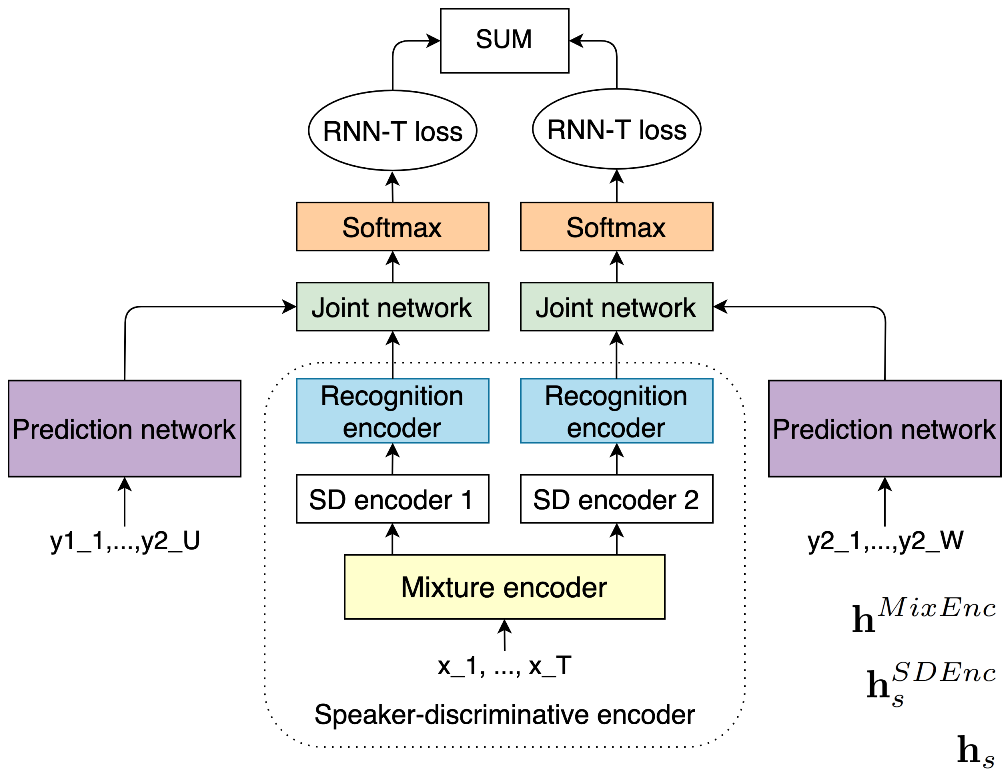
DAT: Deterministic assignment training

PIT: Permutation invariant training

amazon alexa

Multi-speaker RNN-T (MS-RNNT)

DAT-MS-RNN-T



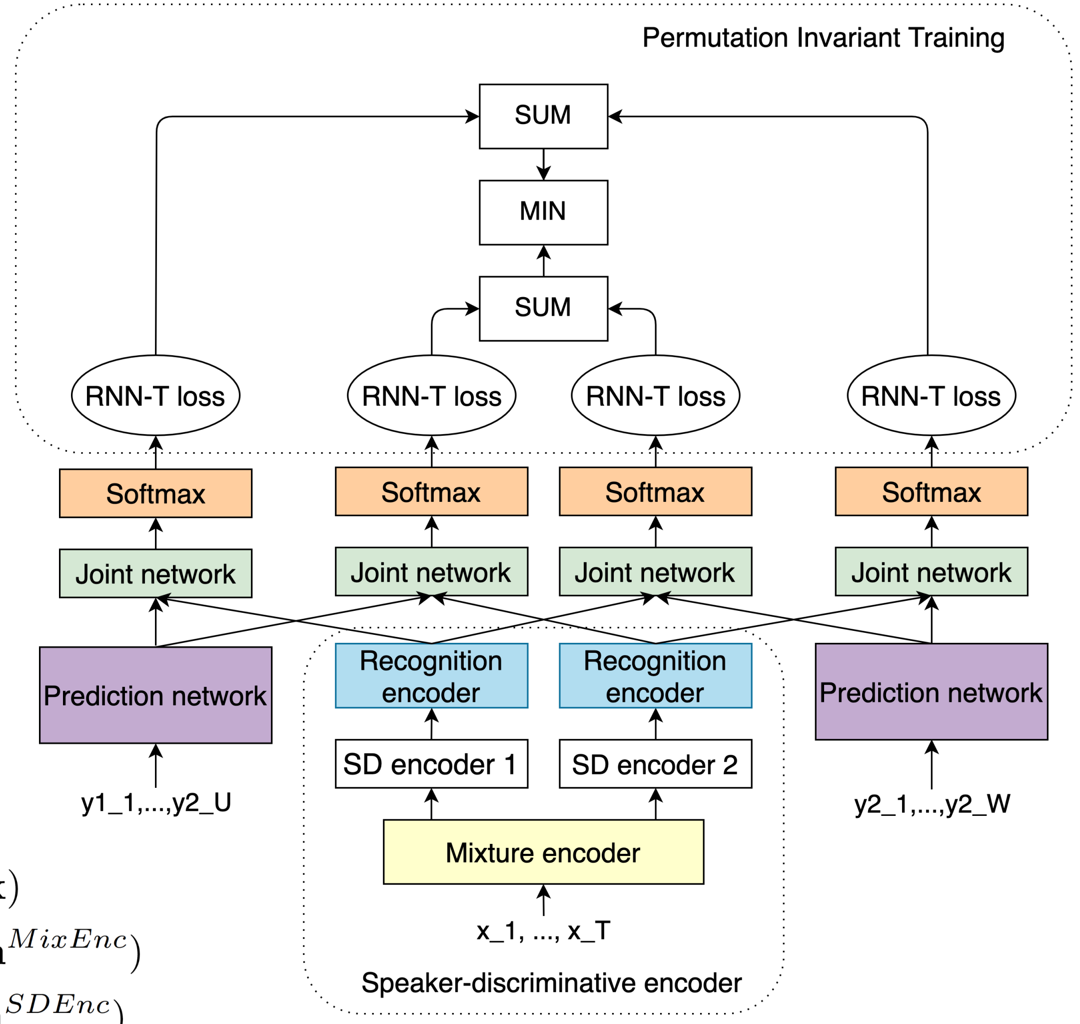
$$\mathcal{L}_{DAT} = -\sum_{s} \log P(\mathbf{y}_s|\mathbf{h}_s)$$

$$\mathbf{h}^{MixEnc} = \mathrm{MixEnc}(\mathbf{x})$$

$$\mathbf{h}_s^{SDEnc} = \mathrm{SDEnc_s}(\mathbf{h}^{MixEnc})$$

$$\mathbf{h}_s = \mathrm{RecEnc}(\mathbf{h}_s^{SDEnc})$$

PIT-MS-RNN-T



$$\mathcal{L}_{PIT} = \min_{\pi \in \mathcal{P}} -\sum_{s} \log P(\mathbf{y}_s|\mathbf{h}_{\pi(s)})$$

DAT: Deterministic assignment training

PIT: Permutation invariant training

amazon alexa

Dataset

- LibriSpeechMix [15]: mixed 2-speaker utterances from LibriSpeech [16]
- Simulation constraints
  - Min. 0.5 sec delay between the speech start of 2 speakers (for train partition only)
  - Each mixture has an overlapping segment
  - Utterances are mixed at 0 dB
- Overall overlap ratios
  - Train: 28%
  - Dev: 25%
  - Test: 24%

amazon alexa

Setup

- Model architecture
  - LSTM encoder with 1024 units, 2 layer per each encoder component
  - LSTM decoder: 2 layers, 1024 units
  - Feed-forward joint network with 1 layer
  - Output vocabulary: 2500 WPs
- Input features
  - 64-dim log-mel filterbanks
  - Frame stacking with a factor of 3
  - Adaptive SpecAugment policy [17]
- Tricks of the trade
  - Pre-training on single-speaker LibriSpeech
  - Multi-style training
  - Speaker order labels [18] as input to SD encoders in DAT-MS-RNN-T

Evaluation

- Optimal edit distance Word Error Rate (WER)
  - Set of permutations (for 2 speakers): $\mathcal{P} = \{(1,2),(2,1)\}$
  - Ground truth $\mathbf{R} = [R_1, ..., R_S]$
  - Model output $\mathbf{O} = [O_1, ..., O_S]$

$$WER = \frac{\min_{(i,j)\in\mathcal{P}}(\sum_{i,j} edits(O_i, R_j))}{\sum_j len(R_j)}$$

amazon alexa

Results

Model	clean	other	2spk	Overall
RNN-T	6.5	15.5	66.3	38.7

Results

- Vanilla DAT achieves 82% WERR on test-2spk w.r.t single-speaker RNN-T
  - WER increase from 6.5% to 9.2% on test-clean due to hypothesis splitting

Model	clean	other	2spk	Overall
RNN-T	6.5	15.5	66.3	38.7
DAT-MS-RNN-T	9.2	16.9	11.8	12.4

amazon alexa

Results

- Vanilla DAT achieves 82% WERR on test-2spk w.r.t single-speaker RNN-T
  - WER increase from 6.5% to 9.2% on test-clean due to hypothesis splitting
- Speaker order labels help to follow the same speaker

Model	clean	other	2spk	Overall
RNN-T	6.5	15.5	66.3	38.7
DAT-MS-RNN-T	9.2	16.9	11.8	12.4
+ speaker order label	7.7	16.2	11.7	11.8

amazon alexa

Results

- Vanilla DAT achieves 82% WERR on test-2spk w.r.t single-speaker RNN-T
  - WER increase from 6.5% to 9.2% on test-clean due to hypothesis splitting
- Speaker order labels help to follow the same speaker
- Multi-style training improves generalization

Model	clean	other	2spk	Overall
RNN-T	6.5	15.5	66.3	38.7
DAT-MS-RNN-T	9.2	16.9	11.8	12.4
+ speaker order label	7.7	16.2	11.7	11.8
+multi-style	7.5	15.4	11.0	11.2

amazon alexa

Results

- Vanilla DAT achieves 82% WERR on test-2spk w.r.t single-speaker RNN-T
  - WER increase from 6.5% to 9.2% on test-clean due to hypothesis splitting
- Speaker order labels help to follow the same speaker
- Multi-style training improves generalization
- Overall performance of PIT-MS-RNN-T is 4% relatively better than DAT-MS-RNN-T

Model	clean	other	2spk	Overall
RNN-T	6.5	15.5	66.3	38.7
DAT-MS-RNN-T	9.2	16.9	11.8	12.4
+ speaker order label	7.7	16.2	11.7	11.8
+multi-style	7.5	15.4	11.0	11.2
PIT-MS-RNN-T	7.9	15.8	10.6	11.2
+multi-style	7.6	15.2	10.2	10.8

Results

- On-par performance on test-2spk with SOT-AED model w/o speaker inventory
  - Fewer parameters
  - Streaming-capable with algorithmic latency of 30ms (feature frame rate)

Model	#params	#speakers in training	clean	2spk
PIT-AED[12]	160.7M	1,2	6.7	11.9
SOT-AED [12]	135.6M	1,2,3	4.6	11.2
SOT-AED[13]	135.6M	1,2,3	4.5	10.3
+ speakerID	145.5M	1,2,3	4.2	8.7
PIT-MS-RNN-T	80.9M	1,2	7.6	10.2

Conclusions and outlook

- Proposed a novel multi-speaker RNN-T model architecture which can be directly applied in streaming applications
  - On-par algorithmic latency with single-speaker RNN-T
- Benchmarked on artificially mixed partially overlapping speech task
  - On par result with non-streaming SOT model
- Investigated single-speaker performance of a multi-speaker model
- Future work
  - Improve robustness to errors on single-speaker data
  - Test on real data (LibriCSS, AMI, CHiME-6, etc.)
  - Generalize to ambiguous number of speakers during inference (1 to N)

THANK YOU

amazon alexa

References

1. *Y. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-channel multi-speaker separation using deep clustering," Interspeech, Sep 2016*
2. *T. Menne, I. Sklyar, R. Schluter, and H. Ney, "Analysis of deep clustering as preprocessing for automatic speech recognition of sparsely overlapping speech," Interspeech, Sep 2019.*
3. *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*
4. *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*
5. *J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," ICASSP, Mar 2016*
6. *Y. Luo and N. Mesgarani, "TaSNet: Time-domain audio separation network for real-time, single-channel speech separation," ICASSP, Apr 2018*
7. *D. Yu, M. Kolbaek, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speakerindependent multi-talker speech separation," ICASSP, Mar 2017*
8. *D. Yu, X. Chang, and Y. Qian, "Recognizing multitalker speech with permutation invariant training," Interspeech, Aug 2017*
9. *Y. Qian, X. Chang, and D. Yu, "Single-channel multitalker speech recognition with permutation invariant training," Speech Communication, vol. 104, p. 1–11, Nov 2018*

References (continue)

10. *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 Long Papers), 2018*
11. *X. Chang, Y. Qian, K. Yu, and S. Watanabe, "End-to-end monaural multi-speaker ASR system without pretraining, ICASSP, May 2019*
12. *N. Kanda, Y. Gaur, X. Wang, Z. Meng, and T. Yoshioka,"Serialized output training for end-to-end overlapped speech recognition," Interspeech, Oct 2020*
13. *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*
14. *A. Graves, "Sequence transduction with recurrent neural networks," CoRR, vol. abs/1211.3711, 2012*
15. *https://github.com/NaoyukiKanda/LibriSpeechMix*
16. *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*
17. *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*
18. *A. Tripathi, H. Lu, and H. Sak, "End-to-end multi-talker overlapping speech recognition," in ICASSP, 2020, pp. 6129–6133*

amazon alexa

amazon

amazon alexa