Multi-stage diarization refinement for the CHiME-7 DASR scenario

Christoph Boeddeker, Tobias Cord-Landwehr, Thilo von Neumann, Reinhold Haeb-Umbach

Paderborn University, Germany

{boeddeker,cord,vonneumann,haeb}@nt.upb.de

Abstract

This submission for the CHiME-7 DASR challenge consists of a TS-VAD system for diarization followed by a GSS system for source extraction. Then, a segment-level refinement is applied to the enhanced audio segments, before using the baseline ASR system for transcribing the audio. As initialization for the TS-VAD, the baseline diarization system was used to identify single-speaker regions that are used to extract enrollment embeddings for each speaker in a meeting. The TS-VAD system is applied on each microphone channel independently, and the soft estimates at the TS-VAD output are averaged across the microphones, before converting them to hard estimates, i.e., the diarization estimates Additionally, we analyzed the estimates and found many speaker swaps and less ideal segments. To address them, we propose a simple post-processing step by comparing speaker embeddings from the baseline diarization, i.e., the enrollment embeddings, with speaker embeddings derived from the enhanced data. Through the usage of TS-VAD, we improve upon the baseline word error rate on the CHiME-6 dataset by 3.6 percentage points, whereas the postprocessing results in an additional consistent word error rate improvement of 2 % to 4 % absolute.

Index Terms: speech recognition, meeting transcription

1. Introduction

The transcription of natural conversations in domestic environments is a challenging task addressing all common speech processing modules ranging from diarization over speech enhancement and automatic speech transcription. In addition, distributed settings require an effective channel/array selection or fusion. Over the last iterations of the CHiME challenge [1–3], diarization, i.e., the task of determining who spoke when, has become increasingly important for the construction of effective meeting transcription systems. During the 5th CHiME challenge [1], the human annotations were available as external diarization. Here, it was shown that using the speaker activity as a guide for source separation [4] leads to a robust and effective speech enhancement that does not require training data. Subsequently, the 6th iteration of CHiME [2] made use of this Guided Source Separation (GSS) [4] in the baseline system and prohibited the usage of the human annotations in a new track. Therefore, inferring an effective diarization from the meeting data became imperative to achieve good results. However, classical diarization pipelines could not cope well with the dynamic conditions of the CHiME meetings, especially for regions of overlapping speech, so no effective guide could be provided by these systems. Here, the newly proposed TS-VAD [5] was able to alleviate these issues and obtain impressive results by performing a diarization refinement using these classical, speaker embedding-based diarization pipelines as initialization. First, a speaker embedding-based diarization pipeline was used to obtain an initial diarization estimate. Then, i-Vectors [6] were extracted from the regions attributed to each speaker to obtain speaker representatives. The TS-VAD system itself then performs a frame-wise classification for each time frame and speaker using the speaker representatives as auxiliary information. While modern neural network-based speaker embeddings such as d-Vectors and x-Vectors [7, 8] significantly outperform i-Vectors [9] in terms of the quality of the speaker representations, the original TS-VAD was unable to make use of this information. This drawback was addressed in [10] for the VoxSRC diarization challenge [11]. However, here only small segments containing two speakers at most were processed by the TS-VAD in order to perform a refinement of overlapping speech for single-channel audio data.

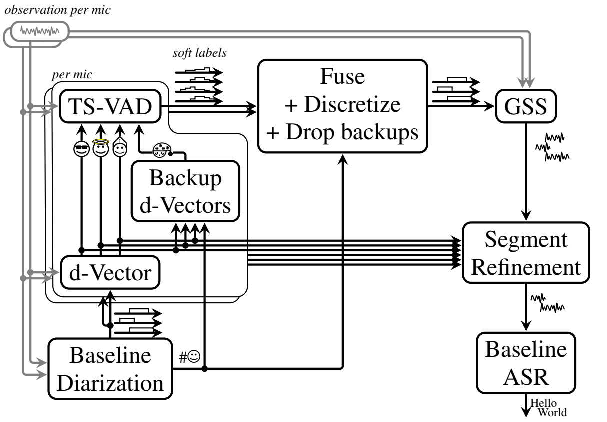

Figure 1: *System overview for the complete processing pipeline.*

In this submission, we only address the diarization task of the CHiME-7 scenario [3] and implement a d-Vector-based TS-VAD for the diarization of the CHiME-7 scenarios to achieve robust diarization results on all subsets of the CHiME-7 data. In addition, we present post-processing steps to combine multiple microphone channels and refine the diarization estimates of the TS-VAD. First, a channel fusion using the soft-decision labels of the neural network, i.e. the TS-VAD, is performed to obtain a channel-independent diarization estimate. Then, these diarization estimates are used as a guide for a GSS system [4]. Since the enhanced segments after GSS might still be corrupted or assigned to the wrong speaker, we propose an additional segment-level refinement that uses segment-level d-Vectors to mitigate these errors. We are able to show that both the CHiME-7 diarization and our proposed diarization pipeline benefit from these postprocessing steps. Here, while maintain-

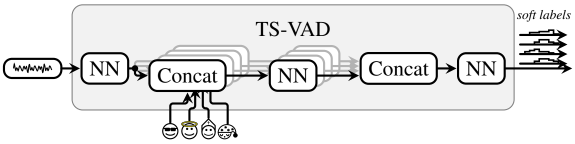

Figure 2: *Illustration of the TS-VAD module*

ing the baseline ASR system, the WER is improved by 1 % to 3 % absolute through the diarization postprocessing steps alone. By replacing the diarization model with a stronger TS-VAD-based diarization, the WER is then further improved.

This work is structured as follows: First, the complete system structure and its added components are described in Section 2. Then, the impact of the modified diarization components and diarization refinement steps on the system performance is investigated in Section 3. Section 4 then shows the final results of our submitted system. We finish this report with a conclusion in Section 5.

2. System Description

The system is a combination of the baseline diarization system, TS-VAD [5], channel fusion, GSS [4], segment-level refinement, and the baseline Automatic Speech Recogniton (ASR) as visualized in Fig. 1.

2.1. TS-VAD

Our implementation from [12] was used as a starting point for the TS-VAD component. The main idea of TS-VAD is to perform a personal Voice Activity Detection (VAD) [13] simultaneously for all active speakers in a meeting. In order to do this, enrollment speaker embeddings are used as auxiliary information to re-identify each speaker. The TS-VAD architecture is split into three parts: a general feature encoder, a speaker-biased module, and lastly a combination layer. The feature encoder only uses the observation as input. Then, the enrollment embedding of one speaker is concatenated to the feature encoder output and fed through the speaker-biased module. This is done for each of the K speakers in a meeting independently to obtain K outputs. These are then concatenated and processed by the combination layer to obtain the diarization estimates for all K speaker simultaneously. A visualization of this processing is in Fig. 2. We use TS-VAD with the observation from one microphone at the input. To process multiple microphones, TS-VAD is applied independently to each microphone, as indicated in Fig. 1.

2.1.1. Enrollment embedding: d-Vector

As enrollment embeddings for the TS-VAD, we used d-Vectors instead of i-Vectors as it was done in [5,12,14]. The main reason for this decision is the embeddings' generalizability to different domains. In [12] it was shown, that a domain adaptation on the input features and the i-Vector embeddings was necessary for a good performance. While i-Vectors need to be carefully adapted so that the domain is not encoded in the embeddings, d-Vectors tend to be less sensitive to these effects. Therefore, we used the neural network-based d-Vectors as enrollments, which have already been shown to work for a TS-VAD system deployed for dialogue data in [10].

At training time, we estimated the enrollments by concatenating the non-overlapping regions per speaker and then extracting a single-speaker embedding for this concatenated data. For the CHiME-6 training data, we used the forced alignments as speaker activity and excluded the overlap according to the human annotations. For SimLibriCSS we had only the simulation annotations, i.e., the utterance boundaries, and used them for both.

During inference time, the human/simulation annotations and the forced alignment are replaced with the CHiME-7 baseline diarization estimates. If a speaker had no non-overlapping regions (which occurred for some Mixer 6 sessions), this speaker was discarded entirely.

2.1.2. Backup embeddings

Since the final combination layer of TS-VAD processes all speakers simultaneously to obtain the diarization estimates, the total number of speakers is fixed. In order to handle scenarios with varying numbers of speakers, [14] proposed to train the model for the maximal number of speakers that can occur. At test time, additional backup enrollment d-Vectors are sampled if fewer speakers are detected in a meeting. The backups are later removed from the TS-VAD output since it is known that they are inactive.

In contrast to [14], we sampled the backups from the development dataset and not from the training dataset. This is done independently for each microphone, because then it is more likely that the average posterior probability (see Section 2.2) is small for the backup speakers, and they have a smaller influence on the desired speakers.

2.2. Channel fusion

The challenge rules prohibit the usage of any prior knowledge about the array geometry for the evaluation. This includes selecting a reference or random microphone, or utilizing which microphones belong to the same array. To address this, we applied the TS-VAD system independently to each microphone channel. This includes the d-Vector extraction, so a slightly different d-Vector is used for each channel. To fuse the channels, we tried two fusion strategies: mean fusion using the soft estimates, and DOVER-LAP [15] on the hard estimates. We use here mean and not median aggregation as in [12], because it slightly improved the Diarization Error Rate (DER) on the Dev data.

2.3. Discretize

For the conversion of the soft estimates to hard estimates, we used the same processing as in [12, 16]: Thresholding and morphological closing (dilation followed by erosion), where the kernel size of dilation is larger to yield overestimates, see Fig. 3.

2.4. GSS

For the source extraction, we used GSS from [4]. It takes the diarization as a guide for potential speaker activity to train a spatial mixture model [17] for each estimated segment on the WPE [18, 19] dereverberated multi-channel observation. The estimated time-frequency posterior is then used for mask-based beamforming.

2.5. Segment-level refinement

While TS-VAD can already be interpreted as a refinement of the baseline diarization system, we here propose a second refinement stage of the diarization that tries to fix speaker confusions and drop noisy segments after the extraction.

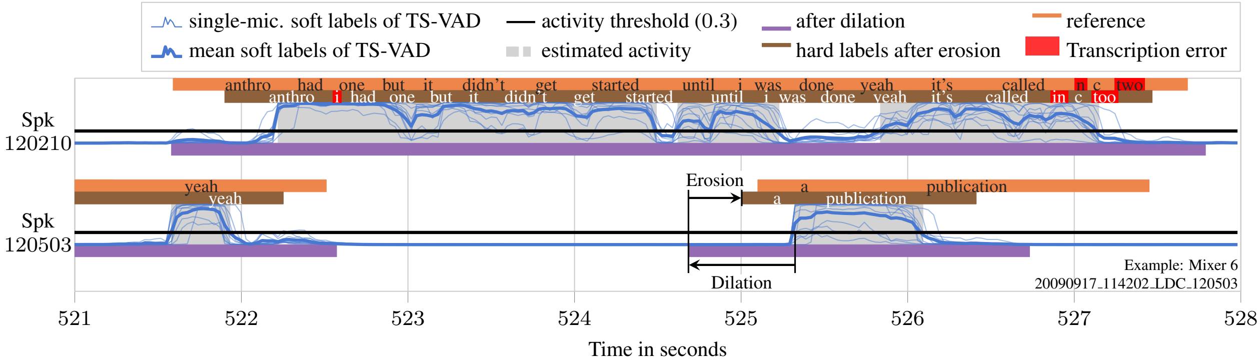

Figure 3: *Illustration of the diarization postprocessing steps and conversion of the soft labels to hard labels, for a section of a Mixer 6 session with mean channel fusion.*

Algorithm 1 d-Vector-based Segment Refinement

```
$\mathbf{d}_{m,k}$ ... d-Vector from concatenated overlap-free speaker regions of the observation for speaker $k$ on mic $m$
$\hat{\mathbf{d}}_b$ ... d-Vector of enhanced speech with target speaker $\hat{k}_b$ in segment $b$
$T_b$ ... Length of segment $b$
for each $b$ do                                              ▷ for each segment
    for each $m$ and $k$ do                                  ▷ for each mic and spk
                                                             ▷ cosine distance
        $c_{b,m,k} = 1 - \frac{\mathbf{d}_{m,k}^{\mathsf{T}} \hat{\mathbf{d}}_b}{\|\mathbf{d}_{m,k}\| \|\hat{\mathbf{d}}_b\|} \in [0, 1]$
    $\tilde{k} \leftarrow \arg\min_{k \neq \hat{k}_b} \left\{ \operatorname{mean}_m \{c_{b,m,k}\} \right\}$      ▷ candidate spk label
    if $\min_m \left\{ c_{b,m,\hat{k}_b} \right\} - 0.05 > \operatorname{mean}_m \left\{ c_{b,m,\tilde{k}} \right\}$ then
        if $\operatorname{mean}_m \left\{ c_{b,m,\tilde{k}} \right\} > 0.6$ then
                                   ▷ assume inactivity, e.g., GSS removed speech
            Drop segment $b$
        else
            $\hat{k}_b \leftarrow \tilde{k}$                 ▷ change speaker label of the segment
    else
        Keep $\hat{k}_b$ as estimated speaker for the segment $b$
$T_k = \sum_b \begin{cases} T_b & \text{if } k = \hat{k}_b, \\ 0 & \text{otherwise.} \end{cases}$
for each $\hat{k}$ do                                        ▷ for each spk
    if $T_k / T^{\text{Recording duration}} < 3\%$ then
                                   ▷ assume inactivity, e.g., speaker counting error
        Remove speaker $k$
```

First, speaker embeddings are extracted for each enhanced segment with the d-Vector system. Then, the pairwise similarities are computed between all segment-level embeddings and all enrollment embeddings, which were also used for TS-VAD. If the difference between the estimated speaker and another speaker exceeds a margin of 0.05, the segment's speaker label is changed. In order to favor the original diarization decision made by the TS-VAD, we used the minimum cosine distance for the speaker that was estimated to be active and the mean cosine distance for all other speakers. If a segment's label is changed and, additionally, no distance is smaller than 0.6, the segment is dropped completely from further evaluation.

After this label correction, speakers with a relative speaker activity of less than 3 % over the course of the meeting are omitted entirely, since they are assumed to stem from errors in the speaker counting. The algorithm is shown in Algorithm 1.

3. Experiments

3.1. TS-VAD

For the training of TS-VAD, we used the data from `https://github.com/jsalt2020-asrdiar/jsalt2020_simulate`, which we call SimLibriCSS, and the CHiME-6 training data. We fixed the number of speakers to 4 for SimLibriCSS to match the upper bound of CHiME-7. We augmented the SimLibriCSS data with noise-only regions extracted from the CHiME-6 training data in a similar way as in the CHiME-6 Kaldi recipe [2].

As input features, we used the concatenation of MFCCs and the logarithm of the spectrum plus one ($\log(1 + \text{Spectrum})$). After concatenation, we normalized each frame to zero mean and unit variance, which could be interpreted as instance normalization [20]. For the training, we used a minibatch size of 3.7 hours (448 sections of 30 second duration) and ADAM [21] with a learning rate of 0.001.

3.2. Embedding extractor: d-Vector system

Enrollment and segment-level embeddings are extracted with the d-Vector system from [22]. The d-Vector extractor is a ResNet34-based system trained on VoxCeleb [23] with MUSAN [24] noise as data augmentation and simulated room impulse responses. For training, the AAM-Softmax loss [25] is used. These 256-dimensional speaker embeddings are used both for extraction of enrollment embeddings for the TS-VAD and to realign speaker segments during the segment refinement. A projection layer is added to TS-VAD that reduces the embedding dimension from 256 to 100 to match the size of the commonly used i-Vectors.

3.3. Word error rates

For the evaluation, we used the Concatenated minimum-Permutation WER (cpWER) [2], Time-Constrained minimum-Permutation WER (tcpWER) [26] and diarization-attributed WER (DA-WER) [3]. The cpWER was used in the previous CHiME challenge and the DA-WER was used for ranking in this challenge. The difference between both metrics lies in the permutation between reference and estimate. The cpWER chooses the permutation that minimizes WER whereas the DA-WER minimizes the Diarization Error Rate (DER) [27]. On the

Table 1: *Comparison of soft-label (Mean) and hard-label (DOVER-Lap) fusion on the CHiME-6 Dev dataset*

Mean		Dover-Lap	
DER	cpWER	DER	cpWER
41.09	**58.66**	**39.80**	61.44

CHiME-7 data, we never observed a difference between both metrics, hence we evaluated our system components with the well-known cpWER. The tcpWER is an upper bound of the cpWER and prohibits correct matches and substitutions in the Levenshtein distance if the words are too far apart from each other. As a maximal temporal distance, we chose a collar of 5 s.

3.4. Effect of channel fusion

Table 1 shows the difference in fusing all channels based on the hard decision labels and based on soft labels. For the hard labels, a diarization output is obtained per channel and then all channels are fused with DOVER-Lap [15]. Due to complexity reasons, the Dover-Lap algorithm is applied in two stages. For soft-label fusion, the mean of the TS-VAD outputs for all microphones are taken and then converted into a single diarization estimate. Interestingly, DOVER-Lap performs better for the DER, but worse in terms of cpWER. Therefore, we used the simpler, and computationally less demanding fusion by taking the average of all soft labels.

3.5. Impact of speaker confusions

In order to assess the influence that wrong speaker assignments have on the cpWER, a lower bound is obtained by relabeling the segments with oracle speaker labels. The oracle speaker labels are chosen such that they minimize the cpWER and are obtained with the ORC-WER algorithm [28,29], where the roles of reference and hypothesis are swapped [12].

Table 2 depicts the cpWER and tcpWER for the estimated labels and the oracle labels for both the baseline system and our TS-VAD pipeline. For the cpWER the gap to the oracle labeling is larger than for tcpWER since the oracle labeling is chosen to minimize the cpWER, which yields overoptimistic results due to temporally irrational alignments. This large gap, often bigger than 10 % absolute, motivated us to try correcting mislabeled segments with the proposed segment-level refinement described in Section 2.5.

While this refinement does not close the gap, it yields a consistent improvement across all datasets in both metrics for our system and the baseline. On the TS-VAD estimates, we separately investigated the effect of dropping uncertain segments and segment relabelling. Both techniques have a positive effect when used individually, but a combination led to the best performance.

3.6. System fusion

For the system fusion, we again used the soft label fusion to combine multiple TS-VAD systems. In addition to fusing all microphones of a single model, they are averaged over multiple systems. First, we tried 3 different checkpoints of one training configuration: Checkpoints after 4000, 11 000, and 25 000 training steps. Since we noticed that the TS-VAD is overfitting to the CHiME-6 training data, this was aimed at stabilizing the system performance. Here, one checkpoint was the best according to the validation training loss, while another came from an earlier and the third from a later stage of the training, i.e., where the system already started overfitting to the training data. For this combination, we already saw a significant gain and hence tried to combine them with four additional systems: One additional checkpoint of our best system configuration (15 000 training steps) and additionally one checkpoint from a TS-VAD system without the embedding projection of the d-Vectors, and two checkpoints from a TS-VAD model, where the SNR of the CHiME-6 noise in the SimLibriCSS data was reduced.

Table 2: *cpWER and tcpWER on the Dev datasets for different model configurations*

System	cpWER			tcpWER		
	CHiME-6	DiPCo	Mixer 6	CHiME-6	DiPCo	Mixer 6
Baseline	62.25	58.16	22.53	66.45	62.86	22.96
+ Segment refinement (relabel + drop)	59.82	57.03	22.24	63.56	61.12	22.66
+ Oracle segment labeling	51.06	45.76	20.95	57.09	52.27	21.55
TS-VAD → GSS	58.66	58.89	21.93	61.89	63.03	22.49
+ Segment drop	56.52	58.53	21.58	59.73	62.29	22.11
+ Segment relabel	56.75	57.68	21.18	61.27	62.29	21.75
+ Segment drop	54.79	57.52	20.94	59.02	61.83	21.49
+ Oracle segment labeling	44.78	46.46	19.72	52.88	53.50	20.61
3 System combination	56.94	56.86	21.11	59.85	60.12	21.69
+ Segment refinement (relabel + drop)	52.86	55.97	20.48	56.61	59.32	21.05
7 System combination	54.30	55.67	20.76	57.34	58.41	21.28
+ Segment refinement (relabel + drop)	**51.12**	**54.31**	20.15	**55.01**	**57.28**	20.68
+ Remove inactive speakers	**51.12**	**54.31**	**19.98**	**55.01**	**57.28**	**20.51**
+ Oracle segment labeling	41.63	45.93	18.92	49.35	51.87	19.73

Table 3: *Evaluation metrics of the submitted system*[1]

Scenario	DER		JER		DA-WER	
	Dev	Eval	Dev	Eval	Dev	Eval
CHiME-6	36.27	58.72	37.43	56.47	51.14	72.90
DiPCo	36.01	29.98	39.73	39.68	54.31	48.27
Mixer 6	15.51	12.40	18.61	11.51	19.98	25.79
Macro	29.27	33.70	31.92	35.89	41.81	48.99

4. Submitted System

The fusion of all 7 TS-VAD system estimates was used for our submission to the CHiME-7 DASR challenge, see Table 3. Here, while using the baseline system and only refining the diarization estimate, significant improvements are achieved over the baseline system for all three subsets of the CHiME-7 challenge data. Since only CHiME-6 and LibriSpeech-based data was used for the training of TS-VAD, the absolute improvement is highest for the CHiME dataset with 11.1 % and slightly worse for the DiPCo data on the Dev data. Still, even without using any matching training data, a gain of 3.7 % and 2.5 % is achieved on DiPCo and Mixer 6, respectively. When taking into account the already good baseline performance on Mixer 6, the relative improvement is of the same magnitude as for CHiME data.

5. Conclusion

For the 7th CHiME challenge, we implemented a simple yet effective postprocessing for diarization systems. Here, we first adapted the TS-VAD system to use neural speaker embeddings and handle different scenarios as well as different numbers of active speakers. We were able to show that a simple soft-label channel fusion allows ignoring outliers in the diarization estimates and using an additional d-Vector-based consistency check to realign enhanced speech segments to the correct speaker or remove incomprehensible segments led to further improvement.

6. Acknowledgements

The first experiments were performed while Christoph Boeddeker was an intern at Mitsubishi Electric Research Laboratories (MERL). We thank Aswin Shanmugam Subramanian, Gordon Wichern, and Jonathan Le Roux for the helpful discussions. Christoph Boeddeker was funded by Deutsche Forschungsgemeinschaft (DFG), project no. 448568305. Computational resources were provided by BMBF/NHR/PC2.

[1] Note: Re-executing the system fusion after the submission caused a small deviation for the CHiME-6 Dev data: Submission 51.14 % cpWER, re-execution 51.12 % cpWER. Values for DiPCo and Mixer6 stayed identical.

7. References

[1] J. Barker, S. Watanabe, E. Vincent, and J. Trmal, "The Fifth 'CHiME' Speech Separation and Recognition Challenge: Dataset, Task and Baselines," in *Interspeech*. ISCA, 2018, pp. 1561–1565.

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

[3] S. Cornell, M. Wiesner, S. Watanabe, D. Raj, X. Chang, P. Garcia, M. Maciejewski, Y. Masuyama, Z.-Q. Wang, S. Squartini, and S. Khudanpur, "The CHiME-7 DASR challenge: Distant meeting transcription with multiple devices in diverse scenarios," 2023.

[4] C. Boeddeker, J. Heitkaemper, J. Schmalenstroeer, L. Drude, J. Heymann, and R. Haeb-Umbach, "Front-end processing for the CHiME-5 dinner party scenario," in *CHiME5 Workshop, Hyderabad, India*, vol. 1, 2018.

[5] I. Medennikov, M. Korenevsky, T. Prisyach, Y. Khokhlov, M. Korenevskaya, I. Sorokin, T. Timofeeva, A. Mitrofanov, A. Andrusenko, I. Podluzhny, A. Laptev, and A. Romanenko, "Target-speaker voice activity detection: A novel approach for multi-speaker diarization in a dinner party scenario," *Proc. Interspeech*, pp. 274–278, 2020.

[6] N. Dehak, P. A. Torres-Carrasquillo, D. Reynolds, and R. Dehak, "Language recognition via i-vectors and dimensionality reduction," in *Twelfth annual conference of the international speech communication association*, 2011.

[7] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, and Z. Zhu, "Deep speaker: an end-to-end neural speaker embedding system," *arXiv preprint arXiv:1705.02304*, 2017.

[8] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333.

[9] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2010.

[10] W. Wang, D. Cai, Q. Lin, L. Yang, J. Wang, J. Wang, and M. Li, "The DKU-DukeECE-Lenovo system for the diarization task of the 2021 VoxCeleb speaker recognition challenge," *arXiv preprint arXiv:2109.02002*, 2021.

[11] A. Brown, J. Huh, J. S. Chung, A. Nagrani, D. Garcia-Romero, and A. Zisserman, "VoxSRC 2021: The third VoxCeleb speaker recognition challenge," *arXiv preprint arXiv:2201.04583*, 2022.

[12] C. Boeddeker, A. S. Subramanian, G. Wichern, R. Haeb-Umbach, and J. L. Roux, "TS-SEP: Joint diarization and separation conditioned on estimated speaker embeddings," *arXiv preprint arXiv:2303.03849*, 2023.

[13] S. Ding, Q. Wang, S.-Y. Chang, L. Wan, and I.-L. Moreno, "Personal VAD: Speaker-conditioned voice activity detection," in *Proc. Odyssey The Speaker and Language Recognition Workshop*, 2020, pp. 433–439.

[14] M. He, D. Raj, Z. Huang, J. Du, Z. Chen, and S. Watanabe, "Target-speaker voice activity detection with improved i-Vector estimation for unknown number of speaker," in *Proc. Interspeech*, 2021, pp. 3555–3559.

[15] D. Raj, L. P. Garcia-Perera, Z. Huang, S. Watanabe, D. Povey, A. Stolcke, and S. Khudanpur, "DOVER-Lap: A method for combining overlap-aware diarization outputs," in *IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 881–888.

[16] C. Boeddeker, T. Cord-Landwehr, T. von Neumann, and R. Haeb-Umbach, "An initialization scheme for meeting separation with spatial mixture models," in *Proc. Interspeech*, 2022, pp. 271–275.

[17] N. Ito, S. Araki, and T. Nakatani, "Complex angular central gaussian mixture model for directional statistics in mask-based microphone array signal processing," in *24th European Signal Processing Conference (EUSIPCO)*. IEEE, 2016, pp. 1153–1157.

[18] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, "Speech dereverberation based on variance-normalized delayed linear prediction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1717–1731, 2010.

[19] L. Drude, J. Heymann, C. Boeddeker, and R. Haeb-Umbach, "NARA-WPE: A python package for weighted prediction error dereverberation in numpy and tensorflow for online and offline processing," in *Speech Communication; 13th ITG-Symposium*. VDE, 2018, pp. 1–5.

[20] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[22] T. Cord-Landwehr, C. Boeddeker, C. Zorilă, R. Doddipatla, and R. Haeb-Umbach, "Frame-wise and overlap-robust speaker embeddings for meeting diarization," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[23] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Computer Speech & Language*, vol. 60, p. 101027, 2020.

[24] D. Snyder, G. Chen, and D. Povey, "MUSAN: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[25] Y. Liu, L. He, and J. Liu, "Large margin softmax loss for speaker verification," in *Proc. Interspeech*, 2019, pp. 2873–2877.

[26] T. von Neumann, C. Boeddeker, M. Delcroix, and R. Haeb-Umbach, "MeetEval: A toolkit for computation of word error rates for meeting transcription systems," 2023, accepted for CHiME-2023 Workshop.

[27] NIST. The 2009 (RT-09) Rich Transcription Meeting Recognition Evaluation Plan. [Online]. Available: https://web.archive.org/web/20100606092041if_/http://www.itl.nist.gov/iad/mig/tests/rt/2009/docs/rt09-meeting-eval-plan-v2.pdf

[28] I. Sklyar, A. Piunova, X. Zheng, and Y. Liu, "Multi-turn RNN-T for streaming recognition of multi-party speech," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 8402–8406.

[29] T. von Neumann, C. Boeddeker, K. Kinoshita, M. Delcroix, and R. Haeb-Umbach, "On word error rate definitions and their efficient computation for multi-speaker speech recognition systems," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.