INCORPORATING END-TO-END FRAMEWORK INTO TARGET-SPEAKER VOICE ACTIVITY DETECTION

Weiqing Wang[1], Ming Li[1,2,*]

[1]Department of Electrical & Computer Engineering, Duke University, Durham, NC 27708, USA
[2]Data Science Research Center, Duke Kunshan University, Kunshan 215316, PR China

{weiqing.wang, ming.li369}@duke.edu

ABSTRACT

In this paper, we propose an end-to-end target-speaker voice activity detection (E2E-TS-VAD) method for speaker diarization. First, a ResNet-based network extracts the frame-level speaker embeddings from the acoustic features. Then, the L2-normalized frame-level speaker embeddings are fed to the transformer encoder which produces the initialization of the speaker diarization results. Later, the frame-level speaker embeddings are aggregated to several target-speaker embeddings based on the output from the transformer encoder. Finally, a BiLSTM-based TS-VAD model predicts the refined diarization results. Several aggregation methods are explored, including soft/hard decisions with/without normalization. Results show that E2E-TS-VAD achieves better performance than the original TS-VAD method with the clustering-based initialization.

Index Terms— End-to-end speaker diarization, target-speaker voice activity detection

1. INTRODUCTION

Speaker diarization is the task of partitioning an audio file into homogeneous segments that belong to the same speaker, and it aims to determine "who spoke when" in an audio recording.

A conventional modular speaker diarization system consists of several independent modules. First, some pre-processing techniques are employed, such as speech enhancement [1], speech separation [2] and dereverberation [3]. Next, voice activity detection is applied to remove the non-speech region [4], and the speech regions are split into several short speaker-homogeneous segments by uniform segmentation [5] or speaker change detection (SCD) [6]. Speaker representations like i-vector [7] or x-vector [8] are then extracted from these segments, where PLDA [5] or cosine distance can be used to measure the pairwise similarities between the speaker representations. Finally, these speaker representations can be partitioned into several groups by the clustering algorithms. Besides, some post-processing methods can be employed to refine the clustering-based diarization results, such as overlap detection, VBx resegmentation [9], target-speaker voice activity detection (TS-VAD) [10], and diarization output voting error reduction (DOVER) [11].

One limitation of the conventional speaker diarization system is that it cannot handle the overlapped speech since it assumes that each segment only contains one speaker. Various methods are proposed to solve this problem, such as aforementioned speech separation, overlap detection, and TS-VAD. TS-VAD has been proved to be successful in many noisy domains, such as CHIME6 [12] and DIHARD III [13] challenge. Another limitation is that the modules of the conventional modular speaker diarization system are separately optimized, which needs carefully tuning on each module. Recently, a self-attentive end-to-end neural diarization (SA-EEND) framework has been proposed to solve these two problems in an end-to-end manner [14].

Considering that both TS-VAD and EEND are good at handling the overlapped speech, we try to combine the EEND and TS-VAD models into a single model. We treat the speaker embedding learned by the EEND model as the target-speaker embedding, and the output from the EEND model can be refined by the TS-VAD model. The key is to learn a good representation and obtain a robust target-speaker embedding, which is the connection between the EEND and TS-VAD model. Fig. 1 shows the architecture of the proposed end-to-end TS-VAD (E2E-TS-VAD) model, where the whole model is jointly optimized.

2. RELATED WORKS

2.1. End-to-end neural diarization

Speaker diarization can be formulated as a multi-label classification problem under the framework of end-to-end neural diarization (EEND) [14], where the permutation-invariant training (PIT) loss is employed to avoid permutation ambiguity [15]. The self-attentive EEND (SA-EEND) [14] is one of the state-of-the-art models which shows great performance on the highly overlapped speech data. Later, an encoder-decoder-based attractor (EDA) is introduced to the SA-EEND, which can handle flexible numbers of speakers and further improve the performance [16]. Similar to EEND-EDA, we also generate speaker embeddings from EEND and generate results based on frame-wise embeddings and speaker embeddings. But the difference is that we process these embeddings in a TS-VAD manner to further improve the performance.

For the SA-EEND, considering an acoustic feature sequence $\mathbf{X} \in \mathbb{R}^{T\times F}$ where T is the length and H is the dimension of the acoustic feature, the Transformer encoder extracts the frame-level embeddings $\mathbf{S} \in \mathbb{R}^{T\times D}$ where D is the dimension of the embedding. Next, a linear layer with a sigmoid function predicts the posterior existence probabilities of each speaker, and the PIT loss minimizes the smallest loss of all permutations computed over all speakers.

2.2. Target-speaker voice activity detection

Target-speaker tracking has been successfully applied to many multi-speaker task, including target-speaker automatic speech recognition (TS-ASR) [17],Voice Filter [18], Personal VAD [19] and TS-VAD [10]. In general, a pre-enrolled target-speaker embedding is calculated, and the speech or activity of that speaker is extracted based on that target-speaker embedding.

Considering an acoustic feature sequence $\mathbf{X} \in \mathbb{R}^{T\times F}$, the TS-VAD model first extract the frame-level speaker information $\mathbf{E} \in$

*Corresponding author: Ming Li

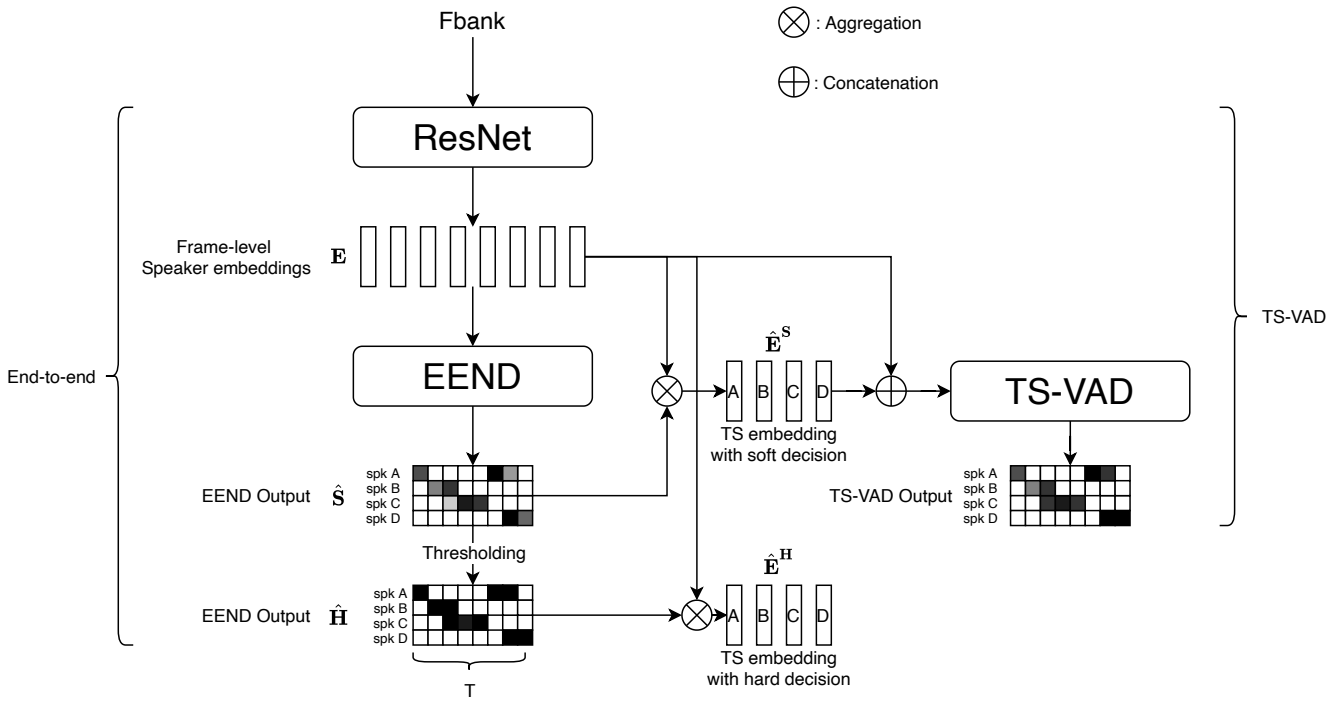

Fig. 1. The architecture of the E2E-TS-VAD model

$\mathbb{R}^{T\times D}$ and several target-speaker embeddings $\mathbf{e}_{\text{target}}$ are concatenated with each frame $\mathbf{e}_i$ separately. Next, several BiLSTM layers extract the speaker detection for each concatenated feature. Finally, the detections of each speaker are concatenated again, and a BiLSTM layer predicts the posterior probabilities for each speaker in frame-level.

3. END-TO-END TARGET-SPEAKER VOICE ACTIVITY DETECTION

3.1. Model architecture

As mentioned in [10], some ideas from EEND are adopted in TS-VAD, e.g., the TS-VAD also predicts speech probabilities for all speakers simultaneously as the EEND does, and they are both optimized by the binary cross-entropy loss function. That is also the reason that we can combine these two models together since they take the same inputs and produce the same outputs. However, the EEND needs to be optimized in a permutation-invariant manner, and TS-VAD requires pre-enrolled target-speaker embeddings. To cope with this, we first apply the PIT loss between the EEND output and label to obtain one optimal permutation of all speakers with the lowest loss. Later, the output can be sort by that permutation, and we aggregate the frame-level embeddings based on the output of the EEND as the target-speaker embeddings. Finally, the target-speaker embeddings are concatenated with the frame-level embeddings as the input of the TS-VAD model, which is the same as the original TS-VAD method in [10].

Fig. 1 shows the model architecture. For the end-to-end model, we employ a ResNet34 to extract the frame-level speaker embedding, and we treat the transformer encoder with the linear layer as the classifier. For the TS-VAD model, we employ the same architecture as [10] where the convolutional layers are removed.

3.2. Speaker embedding aggregation

Suppose that the frame-level speaker embedding sequence $\mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_T]$ produced by the ResNet is L2-normalized and the EEND output before the sigmoid function is $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_1, \hat{\mathbf{y}}_2, ..., \hat{\mathbf{y}}_T]$, where $\mathbf{E} \in \mathbb{R}^{T\times D}$, $\hat{\mathbf{Y}} \in \mathbb{R}^{T\times N}$, T is the length of the sequence, D is the dimension of the embeddings, and N is the number of speakers, the embeddings can be aggregated in several different ways.

3.2.1. Hard decision

We set a threshold for the EEND output $\hat{\mathbf{Y}}$ and obtain the hard decisions for the frame-level speaker embeddings:

$$\hat{\mathbf{H}} = \text{Sigmoid}(\hat{\mathbf{Y}}) \geq \text{threshold}, \tag{1}$$

where $\hat{\mathbf{H}} \in \{0, 1\}^{T\times N}$ is the hard decision matrix whose entry is 1 when the EEND output is great than the threshold and 0 otherwise. Next, the target-speaker embeddings are aggregated by:

$$\hat{\mathbf{E}}^{\text{H}} = \hat{\mathbf{H}}^{\intercal}\mathbf{E}, \tag{2}$$

where $\hat{\mathbf{E}}^{\text{H}} \in \mathbb{R}^{N\times D}$ is the target-speaker embeddings for N speakers. Each target-speaker embedding is divided by the number of active frames of the corresponding speaker in the hard decision matrix. Next, the target-speaker embeddings are L2-normalized and concatenated with the frame-level speaker embeddings.

3.2.2. Soft decision

For the soft decision, we employ two different strategy: linear and softmax. The linear soft decision matrix can be obtained by:

$$\hat{\mathbf{S}} = \text{Sigmoid}(\hat{\mathbf{Y}}), \tag{3}$$

and the softmax soft decision matrix is calculated by applying softmax function along the time axis:

$$\hat{\mathbf{S}} = \text{Softmax}(\hat{\mathbf{Y}}). \tag{4}$$

Finally, similar to Eq. 2, the target-speaker embeddings can be obtained by:

$$\hat{\mathbf{E}}^{\text{S}} = \frac{\hat{\mathbf{S}}^{\intercal}\mathbf{E}}{T}, \tag{5}$$

and $\hat{\mathbf{E}}^{\text{S}}$ is also L2-normalized before concatenation.

3.3. Training objectives

Similar to original EEND [14], we employ the PIT loss [15] for our ResNet-based EEND model:

$$\mathcal{L}_{\text{EEND}} = \frac{1}{TN} \underset{\Phi \in \text{perm}(1,2,\ldots,N)}{\arg\min} \sum_{t=1}^{T} H(\mathbf{y}_t, \text{Sigmoid}(\hat{\mathbf{y}}_t^{\Phi})), \tag{6}$$

where $\mathbf{y}_t$ is the ground-truth label at time step t, $\hat{\mathbf{y}}_t^{\Phi}$ is the permuted output, T is the length of sequence, N is the number of speakers, $\text{perm}(1, 2, ..., N)$ is all permutation sets of all speakers, and $H(\mathbf{y}_t, \hat{\mathbf{y}}_t^{\Phi})$ is the binary cross-entropy loss.

As obtaining the output from the EEND model, we can get the target speaker embeddings as mentioned in Sec. 3.2. The loss function for the TS-VAD model is:

$$\mathcal{L}_{\text{TS-VAD}} = \frac{1}{TN} \sum_{t=1}^{T} H((\mathbf{y}_t), \text{Sigmoid}(\tilde{\mathbf{y}}_t^{\Phi})), \tag{7}$$

where $\tilde{\mathbf{y}}_t^{\Phi}$ is the output of the TS-VAD model. Since the EEND output is already sorted by the permutation Φ, the target-speaker embedding is also sorted, and so does the output of the TS-VAD model.

To further enhance the performance of TS-VAD model, we may add some constrains on the frame-level embedding, e.g., reduce the intra-speaker distance and increase inter-speaker distances as well. Given the L2-normalized frame-level speaker embedding $\mathbf{E} \in \mathbb{R}^{T \times D}$ and target speaker embedding $\hat{\mathbf{E}} \in \mathbb{R}^{N \times D}$, we compute the cosine similarities between them by:

$$\hat{\mathbf{C}} = \mathbf{E}\hat{\mathbf{E}}^{\intercal} \tag{8}$$

where $\hat{\mathbf{E}}$ is the target speaker embedding aggregated by soft or hard decision. Then we employ the multi-label soft margin loss on the cosine similarity matrix $\hat{\mathbf{C}}$:

$$\begin{aligned} \mathcal{L}_{\text{speaker}} = \frac{1}{NT} \sum_{t=1}^{T} \sum_{i=1}^{N} (y_{t,i} * \log(\frac{1}{1 + e^{-\hat{c}_{t,i}}}) \\ + (1 - y_{t,i}) * \log(\frac{e^{-\hat{y}_{t,i}}}{1 + e^{-\hat{c}_{t,i}}})), \end{aligned} \tag{9}$$

where $y_{t,i}$ is the ground-truth label of i$^{\text{th}}$ speaker at time step t, and $\hat{c}_{t,i}$ is the corresponding cosine distance between the i$^{\text{th}}$ target-speaker embedding and the frame-level speaker embedding at time step t. Note that for $\mathcal{L}_{\text{speaker}}$, we can directly use the ground-truth hard decision matrix for target speaker embedding aggregation since it is not required at the testing stage.

Finally, the total training objective is the weight sum of all losses mentioned above:

$$\mathcal{L} = \lambda \mathcal{L}_{\text{EEND}} + (1 - \lambda)\mathcal{L}_{\text{TS-VAD}} + \alpha \mathcal{L}_{\text{speaker}} \tag{10}$$

4. EXPERIMENTS

4.1. Data

The experiments are conducted with the 8 kHz data of two speakers. The training set is simulated on the Switchboard-2 (Phase I, II, III), Switchboard Cellular (Part 1, 2) (SWBD), and the 2004-2008 NIST Speaker Recognition Evaluation (SRE) datasets. Unlike the simulation strategy in the original EEND, we simulate the data with the ground-truth label of the development set, where the audio resources are from SWBD and SRE. The development and evaluation set is the communication telephone speech (CTS) data in the DIHARD III dataset, which contains only two speakers. We directly use the development set for finetuning and the evaluation set for testing. Besides, MUSAN [20] and RIRs [21] corpus are used for data augmentation. The data simulation process and augmentation are applied on the fly during training and finetuning.

4.2. Network parameters

The ResNet34 architecture is similar to that in [22, 23] with the statistical pooling performed at the frame level, and the dimension of the frame-level speaker embedding is $D = 128$. We use two transformer encoder blocks with four heads and 128 attention units and a 128-dim feed-forward layer with a sigmoid function to predict the initialized EEND output. The frame-level embeddings are concatenated with the target-speaker embeddings, producing the $2D = 256$-dim vectors as the input of the TS-VAD model. The first BiLSTM module for speaker detection contains 2 BiLSTM layers with 128 hidden units. The second BiLSTM module contains one layer with 128 hidden units, where the input dimension is $N \times 2D$, and N is the number of speakers. Finally, a linear layer with a sigmoid function predicts the TS-VAD output.

4.3. Training process

During the pre-training stage, we first copy the parameters of a speaker embedding network trained on the 8 kHz VoxCeleb dataset [24] to the ResNet of our E2E-TS-VAD model. Then, we train the network with the simulated data for three steps:

- First, keeping the front-end ResNet frozen, we only train the back EEND and TS-VAD model for 5 epochs to obtain a good initialization. The decision matrix is the ground-truth label for the target-speaker embedding aggregation.
- Second, the front-end ResNet is jointly trained with subsequent models for another 5 epochs. The decision matrix is still the ground-truth label.
- Third, applying different decision matrices mentioned in 3.2 and training the network for 10 epochs.

Finally, we finetune this model on the CTS data of the DIHARD III development dataset for 100 epochs and evaluate it on the test set.

The audio length is 16s for pre-training and finetuning, where silence regions are removed. The acoustic features are the 80-dim filterbank energies with a frame length of 25ms and a frame shift of 10ms. The model is optimized by Adam optimizer. The learning rate is 10^{-4} in the pre-training stage and 10^{-5} in the finetuning stage. For the loss function, λ is 0.5 and α os 0.1. The aggregation strategies are the same for training and inference.

Table 1. The DER (%) on the CTS data of DIHARD III without the speaker loss, where both EEND/TS-VAD DERs are reported.

Decision Type	Aggregation Strategy		L2	DER
	linear	softmax		
Hard	✗	✗	✓	7.66/5.81
	✗	✗	✗	6.96/**5.68**
Soft	✓	✗	✓	9.49/6.61
	✗	✓	✓	8.80/6.76
	✓	✗	✗	7.35/**6.03**
EEND	-	-	-	7.74
Clustering [25]	-	-	-	14.19
TS-VAD [25]	-	-	-	7.03

4.4. Inference and evaluation metric

During the inference stage, we use the whole audio sequence as input with silence regions removed based on the oracle VAD. The EEND output is first obtained and becomes the initialization of the TS-VAD model. After we obtain the first round of EEND and TS-VAD output, we can also use the TS-VAD output as the initialization for the next round of TS-VAD inference, called iterative inference.

After we obtain the posterior probabilities of each speaker, we use a threshold of 0.5 to get the final diarization results. We employ the diarization error rate (DER) as our evaluation metric, where the oracle VAD is used for evaluation. We follow the evaluation protocol in the DIHARD challenge [13], where no forgiveness collar will be applied to the reference segments prior to scoring, and overlapped speech will be evaluated.

Table 2. The DER (%) on the CTS data of DIHARD III with the speaker loss, where both EEND/TS-VAD DERs are reported.

Decision Type	Aggregation Strategy		L2	DER
	linear	softmax		
Hard	✗	✗	✓	6.97/5.75
	✗	✗	✗	7.11/**5.70**
Soft	✓	✗	✓	8.29/6.60
	✗	✓	✓	8.20/6.27
	✓	✗	✗	7.90/**6.17**

4.5. Results

In our experiments, the frame-level speaker embedding are always L2-normalized for aggregation, and we want to find if the L2-normalization has much influence on the target-speaker embedding.

Tab. 1 shows the results of the DERs on the CTS data of the DIHARD III dataset without the speaker loss, where different aggregation strategies are employed. Both the EEND/TS-VAD DERs are reported in Tab. 1. The results show that the hard decision aggregation strategy without L2-normalization achieves the best DER of 5.68%, which is far better than the clustering-based results and also better than the original TS-VAD method of the DIHARD III winner system [25]. The reason is that we have a better initialized diarization results than the clustering-based method does, therefore the TS-VAD model also shows better performance with this better initialization. The TS-VAD can always refine the EEND output and produce a better result if the aggregation strategy is appropriately employed.

In addition, unnormalized target-speaker embedding shows better performance than other decision strategies with L2-normalization. From the results we can know that L2-normalization does not have much importance on the performance, but the aggregation strategy does. We also remove the TS-VAD model and only train the ResNet with the EEND model, result shows that it only achieves a DER of 7.74%. This means that TS-VAD model doesn't only refine the EEND output, but also helps the EEND achieve better performance.

Next, Tab. 2 shows the performance of the speaker loss. It can increase the EEND performance for most systems, but the improvement of TS-VAD is moderate. For unnormalized aggregation strategies, the performance becomes worse. The reason may be that the model can only learn the identity information from the speakers in a recording, which is not enough to learn the discriminative speaker representations.

Besides, we perform iterative inference on all systems, where we use the TS-VAD output as the input of the next round of TS-VAD inference, and we iteratively perform it until the DER converges. Generally, it takes 2 or 3 rounds for convergence. Tab. 3 shows that iterative inference can further improve the performance, especially for the normalized target-speaker embedding compared with Tab. 1.

Table 3. The DER (%) on the CTS data of DIHARD III dataset using iterative inference without the speaker loss, where only TS-VAD DER is reported.

Decision Type	Aggregation Strategy		L2	DER
	linear	softmax		
Hard	✗	✗	✓	**5.64**
	✗	✗	✗	5.66
Soft	✓	✗	✓	6.32
	✗	✓	✓	6.53
	✓	✗	✗	**5.97**

5. CONCLUSION

In this paper, we propose an end-to-end framework for target-speaker voice activity detection (E2E-TS-VAD). Unlike the original TS-VAD method which needs the initialization from the clustering-based results, we use an end-to-end model to obtain the initialized diarization results and feed them to the TS-VAD model in an end-to-end manner. Experimental results also show that the E2E-TS-VAD outperforms the original TS-VAD with clustering-based initialization. Some limitations also exist. First, we do not use the standard EEND framework since it is not compatible with our TS-VAD implementation. Therefore, the EEND performance of the E2E-TS-VAD is worse than the state-of-the-art EEND model. We will extend this method to the standard SA-EEND model and perform experiments on the data with more than two speakers in the future. In addition, speaker loss does not show much improvement in the final diarization results. In the future, we are going to build a global speaker embedding dictionary for our E2E-TS-VAD, which makes it possible to be used for the online speaker diarization.

6. REFERENCES

[1] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 1, pp. 7–19, 2014.

[2] Xiong Xiao, Naoyuki Kanda, Zhuo Chen, Tianyan Zhou, Takuya Yoshioka, Sanyuan Chen, Yong Zhao, Gang Liu, Yu Wu, Jian Wu, et al., "Microsoft Speaker Diarization System for the VoxCeleb Speaker Recognition Challenge 2020," in *Proc. ICASSP 2021*, pp. 5824–5828.

[3] Takuya Yoshioka and Tomohiro Nakatani, "Generalization of multi-channel linear prediction methods for blind mimo impulse response shortening," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 10, pp. 2707–2720, 2012.

[4] Shuo-Yiin Chang, Bo Li, Gabor Simko, Tara N Sainath, Anshuman Tripathi, Aäron van den Oord, and Oriol Vinyals, "Temporal Modeling Using Dilated Convolution and Gating for Voice-activity-detection," in *Proc. ICASSP 2018*, pp. 5549–5553.

[5] Gregory Sell and Daniel Garcia-Romero, "Speaker Diarization with PLDA i-vector Scoring and Unsupervised Calibration," in *Proc. SLT 2014*, pp. 413–417.

[6] Ruiqing Yin, Hervé Bredin, and Claude Barras, "Speaker Change Detection in Broadcast TV Using Bidirectional Long Short-Term Memory Networks," in *Proc. Interspeech 2017*, pp. 3827–3831.

[7] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end Factor Analysis for Speaker Verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2010.

[8] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust DNN Embeddings for Speaker Recognition," in *Proc. ICASSP 2018*, pp. 5329–5333.

[9] Federico Landini, Ján Profant, Mireia Diez, and Lukáš Burget, "Bayesian HMM Clustering of x-vector Sequences (VBx) in Speaker Diarization: Theory, Implementation and Analysis on Standard Tasks," *Computer Speech & Language*, p. 101254, 2021.

[10] Ivan Medennikov, Maxim Korenevsky, Tatiana Prisyach, Yuri Khokhlov, Mariya Korenevskaya, Ivan Sorokin, Tatiana Timofeeva, Anton Mitrofanov, Andrei Andrusenko, Ivan Podluzhny, Aleksandr Laptev, and Aleksei Romanenko, "Target-Speaker Voice Activity Detection: A Novel Approach for Multi-Speaker Diarization in a Dinner Party Scenario," in *Proc. Interspeech 2020*, pp. 274–278.

[11] Andreas Stolcke and Takuya Yoshioka, "DOVER: A Method for Combining Diarization Outputs," in *Proc. ASRU 2018*, pp. 757–763.

[12] Shinji Watanabe, Michael Mandel, Jon Barker, Emmanuel Vincent, Ashish Arora, Xuankai Chang, Sanjeev Khudanpur, Vimal Manohar, Daniel Povey, Desh Raj, David Snyder, Aswin Shanmugam Subramanian, Jan Trmal, Bar Ben Yair, Christoph Boeddeker, Zhaoheng Ni, Yusuke Fujita, Shota Horiguchi, Naoyuki Kanda, Takuya Yoshioka, and Neville Ryant, "CHiME-6 Challenge: Tackling Multispeaker Speech Recognition for Unsegmented Recordings," in *Proc. 6th International Workshop on Speech Processing in Everyday Environments*, 2020, pp. 1–7.

[13] Neville Ryant, Prachi Singh, Venkat Krishnamohan, Rajat Varma, Kenneth Church, Christopher Cieri, Jun Du, Sriram Ganapathy, and Mark Liberman, "The Third DIHARD Diarization Challenge," in *Proc. Interspeech 2021*, pp. 3570–3574.

[14] Yusuke Fujita, Naoyuki Kanda, Shota Horiguchi, Yawen Xue, Kenji Nagamatsu, and Shinji Watanabe, "End-to-end neural speaker diarization with self-attention," in *Proc. ASRU 2019*. IEEE, pp. 296–303.

[15] Dong Yu, Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *Proc. ICASSP 2017*. IEEE, pp. 241–245.

[16] Shota Horiguchi, Yusuke Fujita, Shinji Watanabe, Yawen Xue, and Kenji Nagamatsu, "End-to-End Speaker Diarization for an Unknown Number of Speakers with Encoder-Decoder Based Attractors," pp. 269–273.

[17] Naoyuki Kanda, Shota Horiguchi, Ryoichi Takashima, Yusuke Fujita, Kenji Nagamatsu, and Shinji Watanabe, "Auxiliary Interference Speaker Loss for Target-Speaker Speech Recognition," in *Proc. Interspeech 2019*, pp. 236–240.

[18] Quan Wang, Hannah Muckenhirn, Kevin Wilson, Prashant Sridhar, Zelin Wu, John Hershey, Rif A Saurous, Ron J Weiss, Ye Jia, and Ignacio Lopez Moreno, "Voicefilter: Targeted voice separation by speaker-conditioned spectrogram masking," *arXiv preprint arXiv:1810.04826*, 2018.

[19] Shaojin Ding, Quan Wang, Shuo-yiin Chang, Li Wan, and Ignacio Lopez Moreno, "Personal vad: Speaker-conditioned voice activity detection," *arXiv preprint arXiv:1908.04284*, 2019.

[20] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," *arXiv:1510.08484*, 2015.

[21] Tom Ko, Vijayaditya Peddinti, Daniel Povey, Michael L Seltzer, and Sanjeev Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *Proc. ICASSP 2017*. IEEE, pp. 5220–5224.

[22] Weicheng Cai, Jinkun Chen, Jun Zhang, and Ming Li, "On-the-fly Data Loader and Utterance-level Aggregation for Speaker and Language Recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1038–1051, 2020.

[23] Weicheng Cai, Jinkun Chen, and Ming Li, "Exploring the Encoding Layer and Loss Function in End-to-End Speaker and Language Recognition System," in *Proc. Odyssey 2018*, pp. 74–81.

[24] Arsha Nagrani, Joon Son Chung, Weidi Xie, and Andrew Zisserman, "Voxceleb: Large-scale Speaker Verification in the Wild," *Computer Speech & Language*, vol. 60, pp. 101027, 2020.

[25] Yu-Xuan Wang, Jun Du, Maokui He, Shu-Tong Niu, Lei Sun, and Chin-Hui Lee, "Scenario-Dependent Speaker Diarization for DIHARD-III Challenge," in *Proc. Interspeech 2021*, 2021, pp. 3106–3110.