Speaker Overlap-aware Neural Diarization for Multi-party Meeting Analysis

Zhihao Du and **Shiliang Zhang** and **Siqi Zheng** and **Zhijie Yan**
Speech Lab, Alibaba Group, China
{neo.dzh,sly.zsl}@alibaba-inc.com

Abstract

Recently, hybrid systems of clustering and neural diarization models have been successfully applied in multi-party meeting analysis. However, current models always treat overlapped speaker diarization as a multi-label classification problem, where speaker dependency and overlaps are not well considered. To overcome the disadvantages, we reformulate overlapped speaker diarization task as a single-label prediction problem via the proposed power set encoding (PSE). Through this formulation, speaker dependency and overlaps can be explicitly modeled. To fully leverage this formulation, we further propose the speaker overlap-aware neural diarization (SOND) model, which consists of a context-independent (CI) scorer to model global speaker discriminability, a context-dependent scorer (CD) to model local discriminability, and a speaker combining network (SCN) to combine and reassign speaker activities. Experimental results show that using the proposed formulation can outperform the state-of-the-art methods based on target speaker voice activity detection, and the performance can be further improved with SOND, resulting in a 6.30% relative diarization error reduction.

1 Introduction

Speaker identity is an import information for speaker-attributed automatic speech recognition (Carletta et al., 2005; Barker et al., 2018) and dialogue comprehension (He et al., 2021), especially in multi-party meeting scenarios (Yu et al., 2022a). Recently, speaker diarization is developed to detect and track speakers with acoustic features. In general, speaker diarization methods can be divided into three groups, i.e., clustering-based algorithms, end-to-end models and hybrid systems.

Clustering-based methods comprise three individual parts, including speech segmentation, embedding extraction, and clustering algorithm.

Specifically, a long-term audio is first split into several segments by voice activity detection (VAD). Then, speaker embeddings (Dehak et al., 2011; Wan et al., 2018; Snyder et al., 2018) are extracted for each segment, which are partitioned into clusters by unsupervised clustering algorithms, such as k-means (Dimitriadis and Fousek, 2017), spectral clustering (Ning et al., 2006), agglomerative hierarchical clustering (AHC) (Garcia-Romero et al., 2017), and Leiden community detection (Zheng and Suo, 2022). However, such clustering algorithms are in an unsupervised manner, which does not minimize the diarization errors directly. To overcome this issue, neural models are introduced, such as Li et al. (2021) and Zhang et al. (2022). Recently, the variational Bayesian hidden Markov model (VBx) is introduced to refine the clustering results (Díez et al., 2019). Although VBx achieves impressive performance on several datasets (Castaldo et al., 2008; Ryant et al., 2019), it has trouble handling overlapping speech due to speaker-homogeneous assumption.

End-to-end (E2E) approaches treat speaker diarization as a multi-label classification problem, where a deep neural network is employed to predict a set of binary labels (speaker activities) for each timestep. In this way, E2E models can deal with overlapping speech and minimize diarization errors directly. In Fujita et al. (2019a), the utterance-level permutation-invariant training (uPIT) loss (Kolbaek et al., 2017) is employed to train the end-to-end neural diarization (EEND) model which is further improved in Fujita et al. (2019b) and Liu et al. (2021). To deal with an unknown number of speakers, the encoder-decoder based attractor is involved into EEND (Horiguchi et al., 2020). Another E2E model is the recurrent selective attention network (RSAN), which jointly performs source separation, speaker counting, and diarization. Although a lot of efforts have been made (Horiguchi et al., 2021; Xue et al., 2021; Wang and Li, 2022), E2E mod-

els still have a low scalability to a large number of speakers and long-term audios due to the label permutation problem and memory limitation. To deal with long-term audios in multi-party meeting scenarios, a hybrid system of clustering and neural diarization model can be a reasonable approach (Medennikov et al., 2020; Kinoshita et al., 2021; Yu et al., 2022b). In Medennikov et al. (2020), the unconstrained k-means algorithm is used to extract x-vectors (Snyder et al., 2018) as speaker profiles. Then, the profiles are consumed by a neural diarization model, target-speaker voice activity detection (TSVAD), to obtain frame-level diarization results. TSVAD model takes acoustic features along with x-vectors for each speaker as inputs. A set of binary classification output layers produces activities of each speaker. Recently, TSVAD is widely-used in multi-party meeting scenarios, and achieves the state-of-the-art performance (Yu et al., 2022b; Wang et al., 2022; Zheng et al., 2022).

In both TSVAD and E2E approaches, overlapping speaker diarization is formulated as a multi-label classification problem. This formulation ignores speaker dependency, and speaker overlaps are not explicitly modeled, which can cause much miss detection and false alarm errors. To overcome the disadvantages, we reformulate overlapping speaker diarization as a single-label prediction problem via power set encoding. In this way, speaker dependency and overlaps are explicitly modeled by predicting a single label of speaker combinations rather than a set of multiply binary labels. To fully leverage this formulation, we further propose the speaker overlap-aware neural diarization (SOND) model. In SOND, a context-independent (CI) scorer is employed to model global speaker discriminability, while a context-dependent (CD) scorer is involved to discover local discriminability of speakers in the same audio. Given CI and CD scores, a speaker combining network is proposed to combine and reassign speaker activities. As a results, the proposed method achieves a 6.30% relative improvement than the state-of-the-art methods on a challenging real meeting scenarios [1].

2 System Description

An overview of our speaker diarization system is shown in Figure 1. Given a long-term audio, we first remove its silence regions according to the

results of voice activity detection. Then, we uniformly clip the voiced signal into segments with a fixed length. For each segment, a speaker embedding is extracted from a pre-trained neural network. Next, we perform spectral clustering on the extracted speaker embeddings, and the clustering centroids are treated as speaker profiles. Given the clipped segments and speaker profiles as inputs, a neural diarization model is employed to estimate speaker activities across all timesteps. Finally, we perform post-processing on the diarization results of segments and obtain the transcription of entire long-term audio.

2.1 Voice Activity Detection

Voice activity detection (VAD) aims at finding out the voiced regions in an audio signal. In multi-party meeting scenarios, regions with one or more speakers are marked as "voiced", and others are treated as "unvoiced". In our experiments, VAD results have been already provided by the corpus for fair evaluation, therefore, we directly use the official VAD results in our system.

2.2 Segmentation

For neural diarization model, we uniformly clip the voiced signal with a window size of 16s shifted every 4s. For embedding extraction, to guarantee the speaker-purity of each segment, we further clip the voiced segments into shorter chunks with a chunk size of 1.28s and a shift of 0.64s.

2.3 Embedding Extraction

We employ the ResNet34 (He et al., 2016) as our speaker embedding extractor. The encoding layer is based on statistic pooling (SP), and the dimension of the speaker embedding layer is 256. More details about model architecture are provided in A.1. The ArcFace loss function (Deng et al., 2019) with a margin of 0.25 and softmax pre-scaling of 8 is used to optimize the speaker embedding model.

2.4 Spectral clustering

In our system, we employ a bidirectional long and short term memory (Bi-LSTM) based recurrent neural network (RNN) to construct the affinity matrix for spectral clustering as in Lin et al. (2019). Specifically, m extracted speaker embeddings $[\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_m]$ are concatenated with repeated $\mathbf{e}_i$ to predict the i-th row of affinity matrix. We employ a stacked RNN with two Bi-LSTM layers followed by two fully-connected layers. Each

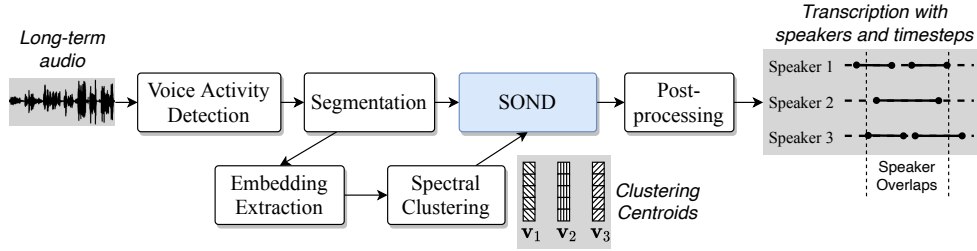[1] Our code is available at https://github.com/ZhihaoDU/du2022sond

Figure 1: An overview of our speaker diarization system. SOND denotes the proposed speaker overlap-aware neural diarization model.

Bi-LSTM layer has 512 units (256 forward and 256 backward). The first fully-connected layer has 64 outputs with the ReLU activation, and the second layer has one unit with the sigmoid activation. After spectral clustering, we obtain the centroids of speakers in the entire long-term audio, which are used as speaker profiles for neural diarization. More details can be found in Lin et al. (2019) and Wang et al. (2022).

It should be noted that the number of speaker profiles can be different among long-term audios, which is not friendly to the following neural diarization. To handle this problem, we zero-pad the profiles of each long-term audio in order to have the same number of profiles N. In our experiments, the maximum number of profiles N is set to 16.

2.5 Neural Diarization

Given T acoustic features of a segment $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T]$ and N speaker profiles of the entire long-term audio $V = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N]$, a neural network (NN) is employed to model the posterior probability $y_{n,t}$, which represents whether speaker n talked at timestep t:

$$P(y_{n,t} = 1|X, V) = \text{NN}(\mathbf{v}_n, \mathbf{x}_t; X, V) \quad (1)$$

We will describe our proposed speaker overlap-aware diarization model in section 3.

2.6 Post-processing

To obtain the entire transcription with speaker and timestep attributes, a two-step post-processing is performed on the diarization results of clipped segments. First, we employ a median filter to smooth the segmental results. According to our preliminary experiments, the filter window size is set to 1.28s. Then, the smoothed segmental results are concatenated in the chronological order to obtain the time stamps for each speaker. We find that the final performance can be further improved by using the predicted transcriptions to extract profiles. Therefore, we run our system three times iteratively.

3 Speaker Overlap-aware Neural Diarization

In the proposed speaker overlap-aware neural diarization model (SOND), input acoustic features and speaker profiles are first projected into the same space by the speech and speaker encoders, respectively. Then, the encoded features and profiles are fed into context-dependent and context-independent scorers. Finally, scores of different speakers are combined to predict the power set encoded labels through a speaker combining network. A diagram of SOND is shown in Figure 2.

3.1 Speech and Speaker Encoders

The speech encoder has the similar model architecture as the embedding extractor, which consists of convolutional blocks (Conv), statistic pooling (SP) and embedding layer (Emb) as described in A.1. However, different from embedding extractor, the statistic pooling of speech encoder is calculated on a window rather then the entire segment:

$$\begin{aligned} \mathbf{h}_t &= \text{SpeechEncoder}(X) \\ &= \text{Emb}(\text{SP}(\text{Conv}(X)\mathbf{I}_{t-2/l:t+l/2})) \end{aligned} \quad (2)$$

where $\mathbf{I}_{t-2/l:t+l/2}$ represents a identity window with ones from $t - l/2$ to $t + l/2$ and zeros otherwise. $\mathbf{h}_t$ denotes the outputs of embedding layer in speech encoder at timestep t. The window length l is set to 20 in our experiments.

The purpose of speaker encoder is to project speaker profiles into the same space of $\mathbf{h}_t$:

$$\bar{\mathbf{v}}_i = \text{SpeakerEncoder}(\mathbf{v}_i) \quad (3)$$

In our experiments, the speaker encoder consists of three fully-connected layers with ReLU activation function, and the output layer has 256 units.
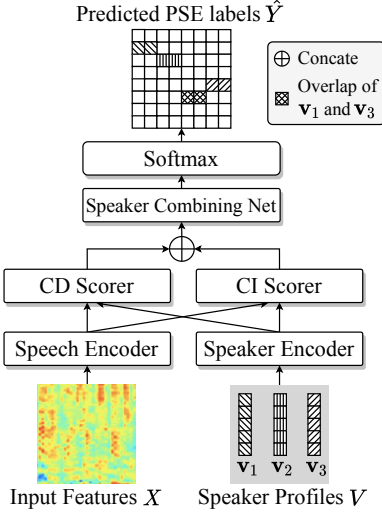
Figure 2: The model diagram of our speaker overlap-aware neural diarization model. CD and CI represent "context-dependent" and "context-independent", respectively.

3.2 Context-independent Scorer

In context-independent (CI) scorer, speakers are detected and tracked through the *global* discriminability, which is learned by contrasting a speaker with others on training set. We employ the cosine similarity of speech encoding $\mathbf{h}_t$ and projected speaker profile $\bar{\mathbf{v}}_i$ to evaluate the probability of speaker n talking at timestep t:

$$\mathbf{S}_{n,t}^{CI} = \frac{<\bar{\mathbf{v}}_n, \mathbf{h}_t>}{||\bar{\mathbf{v}}_n||_2 ||\mathbf{h}_t||_2} \tag{4}$$

where $< \cdot, \cdot >$ represents the inner product of two vectors, and $|| \cdot ||_2$ denotes the L2 norm.

3.3 Context-dependent Scorer

In context-dependent (CD) scorer, speakers are detected and tracked through the *local* discriminability, which is modeled by contrasting a speaker with contextual speakers in the same segment. We employ a multi-head self attention (MHSA) based neural network to predict the context-dependent probabilities of speaker n across all timesteps:

$$
\begin{aligned}
z_{n,0} &= [(\mathbf{h}_1, \bar{\mathbf{v}}_n), (\mathbf{h}_2, \bar{\mathbf{v}}_n), \dots, (\mathbf{h}_T, \bar{\mathbf{v}}_n)] \\
\bar{z}_{n,l} &= z_{n,l-1} + \text{MHSA}_l(z_{n,l-1}, z_{n,l-1}, z_{n,l-1}) \\
z_{n,l} &= \bar{z}_{n,l} + \max(0, \bar{z}_{n,l}W_1^l + b_1^l)W_2^l + b_2^l \\
\mathbf{S}_n^{CD} &= \text{Sigmoid}(W^o z_{n,L^{CD}} + b^o)
\end{aligned} \tag{5}
$$

where $\text{MHSA}_l(Q, K, V)$ represents the multi-head self attention of the l-th layer (Vaswani et al., 2017)

with query Q, key K, and value V matrices. W_*^l and b_*^l denotes the learnable weight and bias of the l-th layer (o for output layer) [2]. Detailed model settings of CD scorer are given in Table 8.

3.4 Speaker Combining Net

To model speaker dependency and overlaps, we propose the speaker combining network (SCN). First, we concatenate CI and CD scores across the speaker axis:

$$
\begin{aligned}
z^0 &= [\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_T] \\
\mathbf{S}_t &= [\mathbf{S}_{t,1}^{CI} \dots, \mathbf{S}_{t,N}^{CI}, \mathbf{S}_{t,1}^{CD}, \dots, \mathbf{S}_{t,N}^{CD}]
\end{aligned} \tag{6}
$$

where N and T are the maximum numbers of speakers and timesteps, respectively. Then, the concatenated scores are combined and reassigned through a feed-forward (FF) projection:

$$
\begin{aligned}
\bar{z}^l &= \text{FF}(z^{l-1}) \\
&= \text{LN}(\max(0, z^{l-1}W_1^l + b_1^l))W_2^l + b_2^l
\end{aligned} \tag{7}
$$

where $W_1^{l=1} \in \mathbb{R}^{2N \times d_{ff}}$, $W_1^{l \neq 1} \in \mathbb{R}^{N \times d_{ff}}$, $W_2^l \in \mathbb{R}^{d_{ff} \times N}$ and the biases $b_1^l \in \mathbb{R}^{d_{ff}}$, $b_2^l \in \mathbb{R}^N$. LN represents the layer normalization (Ba et al., 2016). Subsequently, a sequential memory block (Zhang et al., 2018) is employed to model the activity sequence of speaker n:

$$z_{n,t}^l = \sum_{i=0}^{L_1} a_{n,i}^l \cdot \bar{z}_{n,t-i}^l + \sum_{i=1}^{L_2} c_{n,j}^l \cdot \bar{z}_{n,t+j}^l \tag{8}$$

where a_n^l denotes the weights of historical items looking back to the past with the size of L_1, and c_n^l represents the weights of look-ahead window into the future with the size of L_2. We stack the FF layer and memory block L_{SCN} times, which is followed by a fully-connected layer to predict the probabilities of C power set encoded labels:

$$\hat{\mathbf{y}}_t = \text{Softmax}(W^o z_t^{L_{SCN}} + b^o) \tag{9}$$

where $W^o \in \mathbb{R}^{N \times C}$ and $b^o \in \mathbb{R}^N$. Softmax activation is performed along the category axis. Detailed model settings are given in Table 9.

3.5 Power Set Encoded Labels

In this section, we reformulate overlapping speech diarization as a single-label prediction problem

---

[2]Formally, the weight W of a module should be noted as W^{module}, but for notational simplicity, we omit the superscript of different modules.
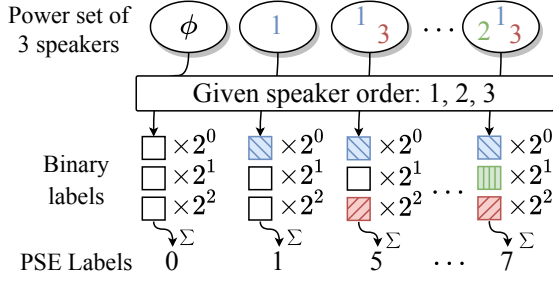
Figure 3: Demonstration of power set encoding with $N = 3$ and $K = 3$. $\sum$ means the summation operator.

through a power set, which can model speaker overlaps explicitly. Given N speakers $\{1, 2, \ldots, N\}$, their power set (PS) is defined as follows:

$$\begin{aligned} \mathrm{PS}(N) &= \{A | A \subseteq \{1, 2, \ldots, N\}\} \\ &= \{\phi, \{1\}, \{2\}, \{1, 2, n, \}, \ldots\} \end{aligned} \quad (10)$$

where ϕ means the empty set. From the definition, we can see that each element of PS represents a combination of speakers, and the power set contains all possible combinations. Therefore, if we treat the elements of PS as classification categories, an overlapping speech frame can be uniquely assigned with a single label. Power-set encoded (PSE) label $\tilde{y}_t$ is obtained by treating the binary label $y_{n,t}$ as an indicator variable:

$$\tilde{y}_t = \sum_{n=1}^{N} y_{n,t} \cdot 2^{n-1} \quad (11)$$

By applying power-set encoding on N speakers, we get 2^N categories, which may be impractical for a large number of speakers. Fortunately, the maximum number of overlapping speakers K is always small (e.g., two, three or four at most) in real-world applications. Therefore, the number of reasonable categories can be reduced to:

$$C = \sum_{k=0}^{K} \binom{N}{k} = \sum_{k=0}^{K} \frac{N!}{k!(N-k)!} \quad (12)$$

In this way, overlapping speech diarization is reformulated as a single-label prediction problem with C categories. Figure 3 shows an example of PSE.

3.6 Training Objective

We adapt a multi-task learning strategy to optimize our SOND model. The main training objective is minimizing the cross entropy loss between

predicted probabilities of PSE labels $\hat{y}_t$ and their ground-truth counterparts $\tilde{y}_t$:

$$\mathcal{J}^{CE}(\theta) = \frac{1}{T} \sum_{t=1}^{T} \mathrm{CrossEntropy}(\hat{y}_t, \tilde{y}_t) \quad (13)$$

where θ denotes learnable parameters of SOND. The second training objective is minimizing the similarity between projected speaker profiles $\bar{v}_n$ in an entire long-term audio:

$$\mathcal{J}^{sim}(\theta) = \sum_{i,j=1}^{N} \max \left(0, \frac{< \bar{v}_i, \bar{v}_j >}{||\bar{v}_i||_2 ||\bar{v}_j||_2} + \delta - 1 \right) \quad (14)$$

where δ represents the expected margin of different speaker profiles. The total training objective is obtained as follows:

$$\theta = \arg \min_{\theta} \mathcal{J}^{CE}(\theta) + \lambda \mathcal{J}^{sim}(\theta) \quad (15)$$

where λ is a hype-parameter to balance the CE and similarity loss. According to our preliminary experiments, we find λ slightly affects the final results, thus we simply set it to 1 in this paper.

4 Experiments

4.1 Alimeeting Dataset

We conduct experiments on the AliMeeting corpus (Yu et al., 2022a), which includes far-field long-term audios recorded by 8-channel microphone array in real meeting scenarios. Since we focus on monaural speaker diarization in this paper, the model training and evaluation are all based on the first-channel data. The training (Train) set of AliMeeting contains 212 audios, about 104.75 hours. The evaluation (Eval) set contains 8 audios (about 4 hours), which are used for model selection and hyper-parameter tuning. The test set (Test) contains 20 audios (about 10 hours), which are employed to evaluate model performance. Each audio consists of a 15 to 30-minute meeting with 2 to 4 speakers. Note that speakers in Train, Eval and Test sets are different from each other. We provide more statistics of AliMeeting in A.3.

4.2 Data for Training Embedding Extractor

We first pre-train the speaker embedding extractor with utterances from the CN-Celeb corpus (Fan et al., 2020). CN-Celeb is a large-scale speaker recognition dataset collected "in the wild". This dataset contains 274 hours from 1,000 celebrities.

To enrich training samples in CN-Celeb, we perform data augmentation with the MUSAN noise dataset (Snyder et al., 2015) and simulated room impulse responses (RIRs) (Ko et al., 2017). We first perform the amplification and tempo perturbation (change audio playback speed but do not change its pitch) on speech. Then, 40,000 simulated RIRs from small and medium rooms are used for reverberation. Finally, the reverberated signals are mixed with background noises at the speech-to-noise rates (SNRs) of 0, 5, 10, and 15 dB.

After pre-training with CN-Celeb, we finetune the embedding extractor on AliMeeting. As the AliMeeting corpus does not provide single-speaker utterances, we select non-overlapping segments according to ground-truth transcriptions, where segments shorter than two seconds are dropped. As a result, we obtain 118,350 utterances from Train set for training and 1,833 utterances from Eval set for parameter selection. The model with lowest equal error rate on Eval set is employed to extract speaker embeddings.

4.3 Data for Training SOND

We first pre-train SOND with a simulated dataset created from the AliMeeting Train set. Details of simulation process are provided in A.4. We run the simulation process 450,000 times to obtain enough training samples. After pre-training with simulated data, we further finetune SOND with real segments from AliMeeting Train set. Segments from the same long-term audio have the same speaker profiles, which contain all speakers talking in the audio. As a result, we obtain 92,569 real training samples. The Eval and Test sets are processed in the same manners as Train set.

4.4 Baselines

We compare our method with VBx and TSVAD (Medennikov et al., 2020). VBx is a clustering-based algorithm, which achieves promising results on several diarization tasks (Landini et al., 2022). We reuse the codes released along with AliMeeting to implement VBx[3]. We adopt the same model architecture as described in Wang et al. (2022) to implement TSVAD. Note that TSVAD is a strong baseline, which achieves the state-of-the-art performance on the AliMeeting corpus (Yu et al., 2022c).

System	Eval	Test
VBx (Díez et al., 2019)	15.24	15.60
IV-TSVAD (Zheng et al., 2022)	5.46	6.92
SC-TSVAD (Wang et al., 2022)	3.49	-
PSE+TSVAD	3.12	4.76
PSE+SOND (Ours)	**2.70**	**4.46**

Table 1: The diarization performance of different methods in terms of DER (%) on the AliMeeting Eval and Test sets. Best results are highlighted by boldface.

4.5 Experimental Settings

We implement and train our models with Tensor-Flow 1.12. Input features are generated by 80-dimensional Mel-frequency filter-banks on each frame, with a window size of 25ms shifted every 10ms. Models are trained with Adam optimizer. The maximum number of overlapped speakers K is set to 4. As a result, PSE labels have 2,517 possible categories.

The embedding extractor is first pre-trained for 200,000 steps on CN-Celeb with the learning rate of 1e-4 and the batch size of 64. Then, we finetune it for another 100,000 steps on AliMeeting with the learning rate of 1e-5.

The training procedure of SOND has three stages. At the first stage, the speech encoder is frozen and initialized with the pre-trained parameters of embedding extractor. We use the simulated data to train the remaining parameters for 200,000 steps with an initial learning rate of 1.0 and 10,000 warm-up steps. At the second stage, we unfreeze the speech encoder and train the whole SOND model on simulated data for 400,000 steps with a fixed learning rate of 1e-4. At the third stage, we employ real training data to finetune the whole model for 50,000 steps with a fixed learning rate of 1e-5. The batch size is set to 32 for all stages. Five models with the best performance on Eavl set are averaged and evaluated on the Test set.

Performance are measured by diarization error rate (DER) (Fiscus et al., 2006), which is a commonly-used metric for speaker diarization tasks [4]. See A.5 for more details of DER.

5 Results

Table 1 shows the diarization performance of different methods in terms of DER on Eval and Test

---

[3]Code is available at https://github.com/yufan-aslp/AliMeeting

[4]The toolkit of DER calculation is available at https://github.com/nryant/dscore.

Model ID	K	C	Eval	Test
SOND-K1	1	17	15.69	16.34
SOND-K2	2	137	4.06	6.15
SOND-K3	3	697	2.97	4.63
SOND-K4	4	2517	2.70	4.46

Table 2: Hyper-parameter tuning on the maximum number of overlapped speakers K. The performance is measured by DER(%) on Eval and Test set. C denotes the number of PSE categories.

Model ID	Margin δ	Eval	Test
SOND-S	0.0	2.79	5.64
SOND-M	0.5	2.73	4.79
SOND-L	1.0	2.70	4.46

Table 3: Hyper-parameter tuning on the similarity margin δ. The performance is measured by DER(%) on Eval and Test set.

Model	MD	FA	SC	DER
SOND	2.29	1.17	1.02	4.46
− Spk. Enc.	2.64	1.55	1.26	5.43
− CD scorer	2.49	1.90	1.88	6.25
− CI scorer	2.39	1.13	1.09	4.60
− PSE	2.29	1.78	0.95	5.00
− SCN	2.56	1.90	1.42	5.88

Table 4: Ablation study on components of SOND. Results are measured in percentage (%). MD, FA and SC denote errors of miss detection, false alarm and speaker confusion, respectively. Spk. Enc. represents speaker encoder for short.

sets. As expected, VBx achieves the worst performance, which is because the overlap ratio of AliMeeting Test set is very high (42.8%), and VBx can not deal with speaker overlaps. By involving a TSVAD model to handle overlap, IV-TSVAD and SC-TSVAD achieve much better performance than VBx. Our implemented TSVAD baseline outperforms the two methods, although they have similar model architectures. This indicates the effectiveness of proposed formulation and PSE. By replacing TSVAD with the proposed SOND, our system achieves the best performance with 13.46% and 6.30% relative improvements on Eval and Test set.

5.1 Hyper-Parameter Tuning

We perform hyper-parameter tuning on the maximum number of overlapped speakers K, and the results are shown in Table 2. Since the audios of AliMeeting corpus consist of 4 speakers at the most, we evaluate 1, 2, 3 and 4 in this experiment. As expected, SOND-K1 achieves the worst performance on both Eval and Test set, which is because speaker overlap is ignored in SOND-K1 as VBx does. By setting K to 2, speaker overlap are considered and diarization errors are significantly reduced. When K increases, more speaker combinations are encoded, and diarization performance is further improved. According to the results, we set $K = 4$ in all following experiments.

We also evaluate the impact of similarity margin δ in Eq.14. Results of $\delta = 0.0, 0.5$ and 1.0 are

shown in Table 3. Margins larger than 1.0 are excluded, since they indicate two speaker embeddings are negatively correlated, which is not desired here. With the smallest margin, SOND-S achieves the worst performance in terms of DER. By increasing the similarity margin from 0.0 to 0.5, SOND-M achieves a 15.07% relative improvement on Test set. Further increasing the margin to 1.0 brings another 6.89% relative error reduction on Test set (seen in SEND-L). We find that increasing the margin does not much improve the performance on Eval set. The reason is two-fold. First, models are selected with Eval set, which may be over-tuned. Second, the overlap ratio of Test set is higher than that of Eval set, which places more demands on global speaker discriminability.

6 Analysis

6.1 Ablation Study

We conduct the ablation and replacement study to evaluate each component of SOND. Results are shown in Table 4. We find removing speaker encoder leads to a 21.75% relative degradation on DER, which reveals the importance of speaker encoder in overlapped regions. While removing CD scorer from SOND leads to a significant performance degradation, removing CI scorer only causes a slight impact. Replacing PSE labels with binary multi-labels makes the overall DER increase from 4.46% to 5.00%, which reveals the effectiveness of PSE. It is interesting to find that PSE affects FA errors severely. This is because PSE explicitly models the dependency of speakers by encoding their combinations. In this way, unrelated speakers will be excluded for an activated speaker, resulting in much less false alarm errors. The impact of SCN is shown in the last row of Table 4. This

Model	Layers	Units	Size(M)	DER(%)
SCN	6	512	19.08	4.46
None	-	-	14.35	5.88
FCN	6	1024	20.45	5.69
CNN	6	512	19.29	5.04
BiLSTMP	4	512	19.30	4.73

Table 5: Diarization performance of different models for speaker combining on Test set.

result reveals the necessity of modeling speaker dependency through combining and reassignment.

6.2 SCN vs. Other Model Architectures

To evaluate the effectiveness of SCN, we compare it with other commonly-used network architectures, such as fully-connected networks (FCN), convolutional neural networks (CNN) and bi-directional LSTM with projection (BiLSTMP). For fair comparison, we keep other components of SOND unchanged and tune model settings to make them have the same size. Table 5 shows the model settings and comparison results. We find removing SCN from SOND causes a significant performance degradation. By employing a FCN to combine frame-level speaker activities, only a slight improvement is obtained. This indicates that frame-level information is not sufficient for speaker combining. Compared with FCN, CNN achieves a much lower DER, which reveals the importance of sequential modeling. To further enhance sequential modeling, BiLSTMP outperforms CNN with a 6.15% relative improvement. With the similar model size, the proposed SCN achieves better performance than BiLSTMP in terms of DER. In addition, outputs of SCN can be calculated in parallel, which is more friendly to modern computers than BiLSTMP.

6.3 Sensitivity to Initial Speaker Profiles

Sensitivity to initial speaker profiles is an important property for neural diarization models in real-world applications. We compare the sensitivity of our SOND and the baseline TSVAD in Table 6, where "Clustering" means profiles are obtained with the results of spectral clustering (SC) and "Oracle" means profiles are extracted using ground-truth transcriptions. For convenience, we also provide the diarization performance of SC in the table, which reflects the quality of speaker profiles.

As expected, models using oracle profiles achieve better performance than those with

Model	Profile Type	Eval	Test	Deg.
SC	-	14.49	14.71	-
TSVAD	Oracle	3.07	4.38	-
TSVAD	Clustering	3.12	4.76	8.68
SOND	Oracle	2.68	4.21	-
SOND	Clustering	2.70	4.46	5.94
− CI scorer	Oracle	3.02	4.18	-
− CI scorer	Clustering	3.03	4.60	10.05
− CD scorer	Oracle	3.31	5.81	-
− CD scorer	Clustering	3.41	6.25	7.57
− Spk. Enc.	Oracle	2.97	4.38	-
− Spk. Enc.	Clustering	2.97	5.43	23.97

Table 6: The sensitivity of models to profile types. Performance is measured by DER(%) on Eval and Test sets. "Deg." represents relative performance degradation on Test set by replacing oracle profiles with clustering-based ones.

clustering-based profiles. The performance gap is more significant on Test set, which is mainly due to the over-tuning issue on Eval set and the higher overlap ratio of Test set. Therefore, we mainly focus on the results of Test set. By replacing oracle profiles with clustering-based ones, an 8.67% relative degradation of TSVAD is observed. For our SOND, using clustering-based profiles leads to a 5.94% relative degradation, which is smaller than TSVAD. This indicates that the proposed SOND is less sensitive to speaker profiles than TSVAD.

To dig out the reason of SOND's robustness to noises in profiles, we further perform an ablation study. From the results, we can see that the speaker encoder and CI scorer are most important components. Removing them causes 23.97% and 10.05% relative degradation. This reveals that global speaker discriminability is crucial for the robustness to noises in profiles.

7 Conclusion and Future Work

In this paper, a hybrid diarization system improved by PSE and SOND is proposed for long-term audios in multi-party meeting scenarios. Specifically, we perform spectral clustering on the neural network based affinity matrix to extract speaker profiles. Then, the profiles are consumed by SOND to predict PSE labels at different time-steps. We find that explicitly modeling speaker dependency and overlaps via PSE labels much reduces diarization errors. In SOND, the local speaker discriminabil-

ity, discovered by CD scorer, is important for final diarization performance. Meanwhile, the global speaker discriminability, modeled by CI scorer, can much improve the robustness to noises in speaker profiles. Compared with other network architectures, SCN is more efficient for combining and re-assigning speaker activities. As a result, our system outperforms the state-of-the-art monaural speaker diarization methods with a 6.30% relative improvement. In the future, we are interested on extending our SOND model to leverage spatial information for multi-channel data.

Limitations

Similar to other neural diarization methods, the proposed method has two main limitations. First, the model architecture of SOND is dependent on the maximum number of speakers in a long-term audio. To deal with variable speaker numbers of different audios, a fixed number N is set and profiles are zero-padded. This can cause extra computational cost, especially when the real speaker number is much smaller than N. Second, power set encoding has a limited scalability to very large number of speakers. According to equation (12), we can see that the number of PSE categories increases exponentially with the maximum number of speakers. When N is small ($\leqslant 16$), setting a maximum number of overlapped speakers can alleviate this problem. For the massively ($N > 16$) multi-party meeting scenarios, this limitation is still an obstacle.

References

Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *CoRR*, abs/1607.06450.

Jon Barker, Shinji Watanabe, Emmanuel Vincent, and Jan Trmal. 2018. The fifth 'chime' speech separation and recognition challenge: Dataset, task and baselines. In *INTERSPEECH*, pages 1561–1565.

Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Maël Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, Guillaume Lathoud, Mike Lincoln, Agnes Lisowska, Iain McCowan, Wilfried Post, Dennis Reidsma, and Pierre Wellner. 2005. The AMI meeting corpus: A pre-announcement. In *MLMI*, volume 3869 of *Lecture Notes in Computer Science*, pages 28–39.

Fabio Castaldo, Daniele Colibro, Emanuele Dalmasso, Pietro Laface, and Claudio Vair. 2008. Stream-based speaker segmentation using speaker factors and eigenvoices. In *ICASSP*, pages 4133–4136.

Najim Dehak, Patrick Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet. 2011. Front-end factor analysis for speaker verification. *IEEE Trans. Speech Audio Process.*, 19(4):788–798.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. 2019. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, pages 4690–4699. Computer Vision Foundation / IEEE.

Mireia Díez, Lukás Burget, Shuai Wang, Johan Rohdin, and Jan Cernocký. 2019. Bayesian HMM based x-vector clustering for speaker diarization. In *INTERSPEECH*, pages 346–350.

Dimitrios Dimitriadis and Petr Fousek. 2017. Developing on-line speaker diarization system. In *INTERSPEECH*, pages 2739–2743.

Y. Fan, J. W. Kang, L. T. Li, K. C. Li, H. L. Chen, S. T. Cheng, P. Y. Zhang, Z. Y. Zhou, Y. Q. Cai, and D. Wang. 2020. Cn-celeb: A challenging chinese speaker recognition dataset. In *ICASSP*, pages 7604–7608.

Jonathan G. Fiscus, Jerome Ajot, Martial Michel, and John S. Garofolo. 2006. The rich transcription 2006 spring meeting recognition evaluation. In *MLMI*, volume 4299 of *Lecture Notes in Computer Science*, pages 309–322. Springer.

Yusuke Fujita, Naoyuki Kanda, Shota Horiguchi, Kenji Nagamatsu, and Shinji Watanabe. 2019a. End-to-end neural speaker diarization with permutation-free objectives. In *INTERSPEECH*, pages 4300–4304.

Yusuke Fujita, Naoyuki Kanda, Shota Horiguchi, Yawen Xue, Kenji Nagamatsu, and Shinji Watanabe. 2019b. End-to-end neural speaker diarization with self-attention. In *ASRU*, pages 296–303.

Daniel Garcia-Romero, David Snyder, Gregory Sell, Daniel Povey, and Alan McCree. 2017. Speaker diarization using deep neural network embeddings. In *ICASSP*, pages 4930–4934.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*, pages 770–778. IEEE Computer Society.

Zihao He, Leili Tavabi, Kristina Lerman, and Mohammad Soleymani. 2021. Speaker turn modeling for dialogue act classification. In *EMNLP (Findings)*, pages 2150–2157. Association for Computational Linguistics.

Shota Horiguchi, Yusuke Fujita, Shinji Watanabe, Yawen Xue, and Kenji Nagamatsu. 2020. End-to-end speaker diarization for an unknown number of speakers with encoder-decoder based attractors. In *INTERSPEECH*, pages 269–273.

Shota Horiguchi, Shinji Watanabe, Paola García, Yawen Xue, Yuki Takashima, and Yohei Kawaguchi. 2021. Towards neural diarization for unlimited numbers of speakers using global and local attractors. In *ASRU*, pages 98–105. IEEE.

Keisuke Kinoshita, Marc Delcroix, and Naohiro Tawara. 2021. Integrating end-to-end neural and clustering-based diarization: Getting the best of both worlds. In *ICASSP*, pages 7198–7202. IEEE.

Tom Ko, Vijayaditya Peddinti, Daniel Povey, Michael L. Seltzer, and Sanjeev Khudanpur. 2017. A study on data augmentation of reverberant speech for robust speech recognition. In *ICASSP*, pages 5220–5224.

Morten Kolbaek, Dong Yu, Zheng-Hua Tan, and Jesper Jensen. 2017. Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks. *IEEE ACM Trans. Audio Speech Lang. Process.*, 25(10):1901–1913.

Federico Landini, Ján Profant, Mireia Diez, and Lukáš Burget. 2022. Bayesian hmm clustering of x-vector sequences (vbx) in speaker diarization: theory, implementation and analysis on standard tasks. *Computer Speech & Language*, 71:101254.

Qiujia Li, Florian L. Kreyssig, Chao Zhang, and Philip C. Woodland. 2021. Discriminative neural clustering for speaker diarisation. In *SLT*, pages 574–581. IEEE.

Qingjian Lin, Ruiqing Yin, Ming Li, Hervé Bredin, and Claude Barras. 2019. LSTM based similarity measurement with spectral clustering for speaker diarization. In *INTERSPEECH*, pages 366–370. ISCA.

Yi-Chieh Liu, Eunjung Han, Chul Lee, and Andreas Stolcke. 2021. End-to-end neural diarization: From transformer to conformer. In *INTERSPEECH*, pages 3081–3085.

Ivan Medennikov, Maxim Korenevsky, Tatiana Prisyach, and et al. 2020. Target-speaker voice activity detection: A novel approach for multi-speaker diarization in a dinner party scenario. In *INTERSPEECH*, pages 274–278.

Huazhong Ning, Ming Liu, Hao Tang, and Thomas S. Huang. 2006. A spectral clustering approach to speaker diarization. In *INTERSPEECH*, pages 2178–2181.

Neville Ryant, Kenneth Church, Christopher Cieri, Alejandrina Cristià, Jun Du, Sriram Ganapathy, and Mark Liberman. 2019. The second DIHARD diarization challenge: Dataset, task, and baselines. In *INTERSPEECH*, pages 978–982.

David Snyder, Guoguo Chen, and Daniel Povey. 2015. MUSAN: A Music, Speech, and Noise Corpus. ArXiv:1510.08484v1.

David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur. 2018. X-vectors: Robust DNN embeddings for speaker recognition. In *ICASSP*, pages 5329–5333.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008.

Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez-Moreno. 2018. Generalized end-to-end loss for speaker verification. In *ICASSP*, pages 4879–4883.

Weiqing Wang and Ming Li. 2022. Incorporating end-to-end framework into target-speaker voice activity detection. In *ICASSP*, pages 8362–8366. IEEE.

Weiqing Wang, Xiaoyi Qin, and Ming Li. 2022. Cross-channel attention-based target speaker voice activity detection: Experimental results for m2met challenge. In *ICASSP*.

Yawen Xue, Shota Horiguchi, Yusuke Fujita, Yuki Takashima, Shinji Watanabe, Leibny Paola García Perera, and Kenji Nagamatsu. 2021. Online streaming end-to-end neural diarization handling overlapping speech and flexible numbers of speakers. In *Interspeech*, pages 3116–3120. ISCA.

Fan Yu, Shiliang Zhang, Yihui Fu, Lei Xie, Siqi Zheng, Zhihao Du, Weilong Huang, Pengcheng Guo, Zhijie Yan, Bin Ma, Xin Xu, and Hui Bu. 2022a. M2met: The icassp 2022 multi-channel multi-party meeting transcription challenge. In *ICASSP*, pages 6167–6171. IEEE.

Fan Yu, Shiliang Zhang, Pengcheng Guo, Yihui Fu, Zhihao Du, Siqi Zheng, Weilong Huang, Lei Xie, Zheng-Hua Tan, DeLiang Wang, Yanmin Qian, Kong Aik Lee, Zhijie Yan, Bin Ma, Xin Xu, and Hui Bu. 2022b. Summary on the ICASSP 2022 multi-channel multi-party meeting transcription grand challenge. In *ICASSP*.

Fan Yu, Shiliang Zhang, Pengcheng Guo, Yihui Fu, Zhihao Du, Siqi Zheng, Weilong Huang, Lei Xie, Zheng-Hua Tan, DeLiang Wang, Yanmin Qian, Kong Aik Lee, Zhijie Yan, Bin Ma, Xin Xu, and Hui Bu. 2022c. Summary on the ICASSP 2022 multi-channel multi-party meeting transcription grand challenge. In *ICASSP*, pages 9156–9160. IEEE.

Chunlei Zhang, Jiatong Shi, Chao Weng, Meng Yu, and Dong Yu. 2022. Towards end-to-end speaker diarization with generalized neural speaker clustering. In *ICASSP*, pages 8372–8376. IEEE.

Shiliang Zhang, Ming Lei, Zhijie Yan, and Lirong Dai. 2018. Deep-fsmn for large vocabulary continuous speech recognition. In *ICASSP*, pages 5869–5873. IEEE.

Naijun Zheng, Na Li, Xixin Wu, Lingwei Meng, Jiawen Kang, Haibin Wu, Chao Weng, Dan Su, and Helen Meng. 2022. The cuhk-tencent speaker diarization system for the ICASSP 2022 multi-channel multi-party meeting transcription challenge. In *ICASSP*, pages 9161–9165. IEEE.

Siqi Zheng and Hongbin Suo. 2022. Reformulating speaker diarization as community detection with emphasis on topological structure. In *ICASSP*.

A   Appendix

A.1   Speaker Embedding Extractor

The architecture details of our speaker embedding extractor is given in Table 7, where T and D denote the sequence length and dimension of input features, respectively. C represents the total number of speakers on the training set. "ds" means down-sampling layer for short.

Layer	Parameters	Output
reshape	-	$T \times D \times 1$
conv1	$3 \times 3, 32, 1$	$T \times D \times 32$
conv2_x	$\begin{bmatrix} 3 \times 3, 32, 1 \\ 3 \times 3, 32, 1 \end{bmatrix} \times 3$	$T \times D \times 32$
ds1	$3 \times 3, 64, 2$	$\frac{T}{2} \times \frac{D}{2} \times 64$
conv3_x	$\begin{bmatrix} 3 \times 3, 64, 1 \\ 3 \times 3, 64, 1 \end{bmatrix} \times 4$	$\frac{T}{2} \times \frac{D}{2} \times 64$
ds2	$3 \times 3, 128, 2$	$\frac{T}{4} \times \frac{D}{4} \times 128$
conv4_x	$\begin{bmatrix} 3 \times 3, 128, 1 \\ 3 \times 3, 128, 1 \end{bmatrix} \times 6$	$\frac{T}{4} \times \frac{D}{4} \times 128$
ds3	$3 \times 3, 256, 2$	$\frac{T}{8} \times \frac{D}{8} \times 256$
conv5_x	$\begin{bmatrix} 3 \times 3, 256, 1 \\ 3 \times 3, 256, 1 \end{bmatrix} \times 3$	$\frac{T}{8} \times \frac{D}{8} \times 256$
fc	$256 \times 256, -, -$	$\frac{T}{8} \times \frac{D}{8} \times 256$
pooling	global statistic pooling	1×512
embedding	$512 \times 256, -, -$	1×256
output	$256 \times M$, softmax	$1 \times C$

Table 7: The architecture of our speaker embedding extractor. Parameter settings are given in the format of "height $\times$ width, channel number, stride".

A.2   Detailed Model settings

The model settings of CD scorer and SCN are given in Table 8 and Table 9.

Name	Value
Layers L^{CD}	4
Attention Dimension	512
Attention Heands	4
Weight Dimension	1024
Output Dimension	1

Table 8: Detailed model settings of CD scorer.

Name	Value
Layers L^{SCN}	6
FF Dimension d_{ff}	512
Look-back Length L_1	15
Look-ahead Length L_2	15

Table 9: Detailed model settings of SCN.

A.3   Details of AliMeeting dataset

We use open-source AliMeeting corpus for our experiments, which is available at https://www.openslr.org/119. Table 10 shows the statistics of the dataset.

Attributes	Train	Eval	Test
Duration (hour)	104.75	4.00	10.00
Sessions	212	8	20
Rooms	12	5	6
Total Speakers	456	25	60
Total Males	246	12	31
Total Females	210	13	29
Overlap Ratio (%)	42.27	34.20	42.8

Table 10: Statistics of AliMeeting corpus.

A.4   Data Simulation Process

The simulated training samples for SOND are created as follows:

1. Select all non-overlapped speech for each speaker in the Train set.

2. Extract the binary labels from the transcriptions and remove the silence regions.

3. Randomly select a continuous segment of binary labels with the duration of 16s and fill the active region with non-overlapped speech segments.

4. Extract speaker profiles for used speakers in this segment.

5. Augment speaker profiles with unused speakers in the training set.

6. Repeat Step 3-5 many times.

A.5 Evaluation Metric

Diarization error rate (DER) is calculated as: the summed time of three different errors of speaker confusion (SC), false alarm (FA) and missed detection (MD) divided by the total duration time:

$$\text{DER} = \frac{\mathcal{T}_{SC} + \mathcal{T}_{FA} + \mathcal{T}_{MD}}{\mathcal{T}_{Total}} \times 100\% \quad (16)$$

where $\mathcal{T}_{SC}$, $\mathcal{T}_{FA}$ and $\mathcal{T}_{MD}$ are the duration of the three errors, and $\mathcal{T}_{Total}$ is the total duration. In order to mitigate the effect of inconsistent annotations and human errors in reference transcriptions, we set a 0.25 second "no score" collar around every boundary of the reference segment.

A.6 Average Runtime

Stage	Training	Inference
Pre-train	1d 17.8h	
Train	3d 22.9h	3min
Finetune	11.4h	

Table 11: Average runtime of SOND.

In Table 11, we list the average runtime using two V100 GPUs of 1) Pre-train: freezing speech encoder and training the remaining parameters on simulated data, 2) Train: training the whole model on simulated data, 3) Finetune: finetuning the whole model on real data. The cost time of inference on Test set is also given in the table.

A.7 Computing Infrastructure

We conduct our experiments on NVIDIA V100 GPU (16GB) and Intel(R) Xeon(R) Platinum 8163 32-core CPU @ 2.50GHz.