INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Online Streaming End-to-End Neural Diarization Handling Overlapping Speech and Flexible Numbers of Speakers

Yawen Xue[1], *Shota Horiguchi*[1], *Yusuke Fujita*[1], *Yuki Takashima*[1], *Shinji Watanabe*[2]
Paola García[3], *Kenji Nagamatsu*[1]

[1] Hitachi, Ltd. Research & Development Group, Japan
[2] Language Technologies Institute, Carnegie Mellon University, USA
[3] Center for Language and Speech Processing, Johns Hopkins University, USA

yawen.xue.wn@hitachi.com

Abstract

We propose a streaming diarization method based on an end-to-end neural diarization (EEND) model, which handles flexible numbers of speakers and overlapping speech. In our previous study, the speaker-tracing buffer (STB) mechanism was proposed to achieve a chunk-wise streaming diarization using a pre-trained EEND model. STB traces the speaker information in previous chunks to map the speakers in a new chunk. However, it only worked with two-speaker recordings. In this paper, we propose an extended STB for flexible numbers of speakers, FLEX-STB. The proposed method uses a zero-padding followed by speaker-tracing, which alleviates the difference in the number of speakers between a buffer and a current chunk. We also examine buffer update strategies to select important frames for tracing multiple speakers. Experiments on CALLHOME and DIHARD II datasets show that the proposed method achieves comparable performance to the offline EEND method with 1-second latency. The results also show that our proposed method outperforms recently proposed chunk-wise diarization methods based on EEND (BW-EDA-EEND).

Index Terms: online speaker diarization, EEND, overlapping speech, flexible numbers of speakers

1. Introduction

Speaker diarization, a challenging technique that responds to the question "who spoke when" [1–7], assigns speaker labels to audio regions. Diarization produces outcomes that downstream tasks can utilize. For example, it can provide the turn-taking information and build a pre-processing pipeline for automatic speech recognition in meetings [8–11], call-center telephone conversations [12–14], and home environments [15–17].

The three challenging aspects that current speaker diarization systems should fulfill are overlapping speech, unknown number of speakers, and online operation. However, it is still an open problem to solve these conditions at once. Conventional clustering-based systems primarily focus on clustering algorithms and speaker embeddings such as Gaussian mixture models (GMM) [18, 19], i-vector [20–22], d-vector [23, 24], and x-vector [25, 26]. However, most clustering-based systems assume that there is only one speaker per segment. As a result, these systems cannot deal with the overlapping speech in general except for a few studies, e.g., [27].

To solve the overlapping issue, an end-to-end neural diarization model (EEND) was proposed [28]. EEND directly minimizes the diarization error by mapping the multi-speaker mixture recording to joint speech activities using a single neural network. The model estimates the speech activity using a dedicated stream for every speaker; hence, EEND inherently assigns two or more labels to the overlapping regions. EEND has already shown significant performance improvement on overlapping speech, especially after adopting the self-attention mechanism (SA-EEND) [29], and with a fixed number of speakers. To deal with overlapping speech and flexible numbers of speakers, Horiguchi *et al.* introduced the encoder-decoder based attractor (EDA) module to SA-EEND [31], and Fujita *et al.* extended the SA-EEND to speaker-wise conditional EEND (SC-EEND) [32, 36]. Both extensions have only been evaluated in offline mode.

Table 1: *Comparison of speaker diarization methods.*

Method	Online	Overlapping	Flexible #speakers
x-vector+clustering [25]	–	–	✓
UIS-RNN [23, 24]	✓	–	✓
EEND/SA-EEND [28–30]	–	✓	–
EEND-EDA/SC-EEND [31, 32]	–	✓	✓
RSAN [33, 34]	✓	✓	✓
BW-EDA-EEND [35]	✓	✓	✓
This work	✓	✓	✓

To cope with online applications, the speaker-tracing buffer (STB) [37] was proposed to trace the speaker permutation information across chunks which enables the offline pre-trained SA-EEND model to work in an online manner. The original STB achieved comparable diarization accuracy to the offline EEND with 1 s chunk size but this method was limited to two-speaker recordings. In [35], Han *et al.* proposed the block-wise-EDA-EEND (BW-EDA-EEND) which makes the EDA-EEND work in an online fashion. Motivated by Transformer-XL [38], this approach utilizes the previous hidden states of the transformer encoder as input to the EDA-EEND.

To satisfy all the three requirements together, among the existing diarization methods as shown in Table 1, the Recurrent Selective Attention Network (RSAN) [33, 34] and the block-wise-EDA-EEND (BW-EDA-EEND) stand out. However, due to the speech separation-based training objective, RSAN is hard to adapt to real recordings, and the evaluations under real scenarios are not reported. On the other hand, although BW-EDA-EEND [35] conducted online experiments on 10 s chunk size conditions, which cause large latency. In this paper, we consider more realistic *streaming* applications with a smaller chunk size such as 1 s.

In this work, we extend the inference algorithm of existing offline model (e.g., EEND-EDA) to operate in an online mode using the speaker-tracing buffer for flexible numbers of speakers (FLEX-STB) without re-training the offline model. FLEX-STB is designed to deal with variable numbers of speak-

http://dx.doi.org/10.21437/Interspeech.2021-708

ers using a zero-padding mechanism with reasonable latency. Four frame selection strategies are also proposed to contain the speaker permutation information in FLEX-STB. The proposed diarization system can operate in an *online mode* handling *overlapping speech* and *flexible number of speakers*, and working in real scenarios such as CALLHOME and DIHARD II with 1 s chunk size.

2. Preliminary

In this section, we briefly explain two key elements: EEND for flexible numbers of speakers and the original STB that enables the offline SA-EEND systems to work online.

2.1. EEND for flexible numbers of speakers

Given a T-length sequence of D-dimensional log-scaled Mel-filterbank-based acoustic features $\mathbf{X} \in \mathbb{R}^{D\times T}$, a neural network-based function $\mathrm{EEND} : \mathbb{R}^{D\times T} \to (0,1)^{S\times T}$ calculates posterior probabilities of speech activities at each time frame $\hat{\mathbf{Y}} = (\hat{\mathbf{y}}_t)_{t=1}^{T} \in (0,1)^{S\times T}$ as follows:

$$\hat{\mathbf{Y}} = \mathrm{EEND}(\mathbf{X}), \tag{1}$$

Here, $\hat{\mathbf{y}}_t := [\hat{y}_{1,t},\ldots,\hat{y}_{S,t}]^\mathsf{T}$ is the posterior of speech activities calculated for each speaker $s \in \{1,\ldots,S\}$ independently, where $(\cdot)^\mathsf{T}$ denotes the matrix transpose and S is the number of speakers. Diarization results $\tilde{\mathbf{Y}} = (\tilde{y}_{s,t})_{s,t} \in \{0,1\}^{S\times T}$ are obtained by applying a threshold value θ (*e.g.*, 0.5) to the posteriors $\hat{\mathbf{Y}}$. If $\tilde{y}_{s,t} = \tilde{y}_{s',t} = 1$ $(s \neq s')$, it means that both speakers s and s' are estimated to have spoken at time t, which is regarded as the overlapping region. If $\forall s \in \{1,\ldots,S\}$, $\tilde{y}_{s,t} = 0$, it indicates that no speaker is estimated to have spoken at time t. Note that EEND used permutation invariant training [28] so that there is no condition to decide the order of output speakers.

While the original EEND [28, 30] fixes the number of speakers S by its network structure, variants of EEND [31, 32, 36] have been proposed to estimate the number of speakers $\hat{S}$. However, these methods perform only in the offline setting.

2.2. Speaker-tracing buffer for fixed number of speakers

One of the straightforward online extensions of EEND is to perform diarization process for each chunk of acoustic features and concatenated diarization results across the chunk. However, this cannot obtain a consistent speaker permutation of the whole recording. This is because the EEND used permutation invariant training [28] so that there is no condition to decide the order of output speakers. We call this speaker permutation problem. To solve the speaker permutation problem, we have proposed speaker-tracing buffer (STB) [37] for the original EENDs, which assume that the number of speakers was known as prior.

Let $\mathbf{X}_i \in \mathbb{R}^{D\times\Delta}$ represents the subsequence of $\mathbf{X}$ at chunk $i \in \{1,\ldots,I\}$ with a fixed chunk length Δ, i.e., $\mathbf{X} = [\mathbf{X}_1,\ldots,\mathbf{X}_i,\ldots,\mathbf{X}_I]$. The $\mathrm{EEND} : \mathbb{R}^{D\times T} \to (0,1)^{S\times T}$ function accepts the input features of flexible length T and produces the posteriors of speech activities of the same length for each speaker. Note that the number of speakers S is fixed in this section.

2.2.1. Initialization

The STB possesses two matrices: acoustic features $\mathbf{X}_i^{(\text{buf})} \in \mathbb{R}^{D\times L_i}$ and the corresponding posteriors $\mathbf{Y}_i^{(\text{buf})} \in \mathbb{R}^{S\times L_i}$ from $\mathrm{EEND}\,(\cdot)$, where L_i is the buffer length after i-th update. The matrices are initialized at the first chunk as follows:

$$\mathbf{X}_1^{(\text{buf})} = \mathbf{X}_1, \tag{2}$$

$$\mathbf{Y}_1^{(\text{buf})} = \hat{\mathbf{Y}}_1 = \mathrm{EEND}(\mathbf{X}_1). \tag{3}$$

As we assume that the chunk size Δ is smaller than the maximum number of frames L_{max} in the buffer, all the inputs and outputs of the first chunk can be fed into STB.

2.2.2. Chunk-wise processing handling speaker permutation

From the second chunk, posteriors $\hat{\mathbf{Y}}_i$ are computed using the STB. Firstly, an input concatenated with the the buffer is fed into $\mathrm{EEND}\,(\cdot)$:

$$\left[\mathbf{Y}_{i-1}^{(\text{buf})}, \hat{\mathbf{Y}}_i\right] = \mathrm{EEND}\left(\left[\mathbf{X}_{i-1}^{(\text{buf})}, \mathbf{X}_i\right]\right) \in (0,1)^{S\times(L_{i-1}+\Delta)}. \tag{4}$$

Next, the optimal speaker permutation for the current chunk is calculated as follows:

$$\psi = \underset{\phi\in\mathrm{Perm}(S_i)}{\arg\max}\ \mathrm{Corr}\left(\mathbf{Y}_{i-1}^{(\text{buf})}, \mathbf{P}_\phi \hat{\mathbf{Y}}_{i-1}^{(\text{buf})}\right), \tag{5}$$

where $\mathbf{P}_\phi \in [0,1]^{S\times S}$ is a permutation matrix for the ϕ-th permutation in $\mathrm{Perm}(S_i)$, which is all the possible permutations of the sequence $(1,\ldots,S)$. $\mathrm{Corr}\,(\mathbf{A},\mathbf{B})$ calculates the correlation between two matrices $\mathbf{A} = (a_{ij})_{jk}$ and $\mathbf{B} = (b_{jk})_{ij}$ defined as

$$\mathrm{Corr}\,(\mathbf{A},\mathbf{B}) := \sum_{i,j} (a_{jk} - \bar{a})\left(b_{jk} - \bar{b}\right), \tag{6}$$

where $\bar{a}$ and $\bar{b}$ are the mean values of $\mathbf{A}$'s and $\mathbf{B}$'s elements, respectively. Finally, the posterior probabilities of the i-th chunk are calculated with the permutation matrix that gives the highest correlation as follows:

$$\mathbf{Y}_i = \mathbf{P}_\psi \hat{\mathbf{Y}}_i. \tag{7}$$

If the length of $\left[\mathbf{Y}_{i-1}^{(\text{buf})}, \mathbf{Y}_i\right]$ is larger than the predetermined maximum buffer length L_{max}, we select frames to be kept in the STB, which are used to solve the speaker permutation problem occurred by the future inputs. In the paper [37], four selection strategies have been proposed.

The STB is a solution to the online diarization problem; however, it cannot be directly applied to EEND for unknown and flexible numbers of speakers. One reason is because the number of speakers may be different across chunks so that we cannot calculate correlation using Eq. (6). The other reason is that the most promising selection strategy used the absolute difference of probabilities of two speakers' speech activities; thus, the method is limited to two-speaker EENDs.

3. Proposed method

In this paper, we proposed the FLEX-STB which extends the STB coping with the two obstacles to use it with EEND for unknown numbers of speakers [31,32]. The FLEX-STB deals with the varying number of speakers across chunks by increasing the number of speaker slots in the speaker-tracing buffer with the zero-padding in Section 3.1. When the system detects new speakers, it adds new zero-speaker-activity slots to the speaker buffer. We also propose four selection strategies to update the buffer, each of which are not limited by the number of speakers, in Section 3.2.

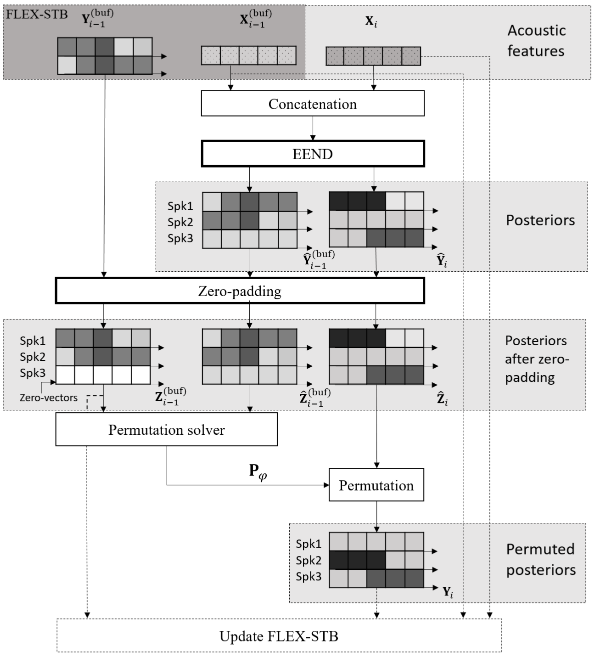

Figure 1: *Proposed speaker-tracing buffer for unknown numbers of speakers. Zero-padding is applied to mitigate the different number of speakers between* $\mathbf{Y}_{i-1}^{(buf)}$ *and* $\hat{\mathbf{Y}}_{i-1}^{(buf)}$.

3.1. Speaker-tracing buffer for flexible numbers of speakers (FLEX-STB)

In this section, we assume that EEND estimates not only speech activities but also the number of speakers S, i.e., EEND : $\mathbb{R}^{D\times T} \to (0,1)^{S\times T}$. Firstly, to alleviate the different number of speakers between the buffer $\mathbf{Y}_{i-1}^{(\mathrm{buf})}$ and the current chunk's output $\hat{\mathbf{Y}}_i$, the posterior of the no-speech-activity speaker is considered as zero so that the zero-padding function is applied as follows:

$$\mathbf{Z}_{i-1}^{(\mathrm{buf})} = \mathsf{ZeroPadding}\left(\mathbf{Y}_{i-1}^{(\mathrm{buf})}, S_i\right), \tag{8}$$

$$\left[\hat{\mathbf{Z}}_{i-1}^{(\mathrm{buf})}, \hat{\mathbf{Z}}_i\right] = \mathsf{ZeroPadding}\left(\left[\hat{\mathbf{Y}}_{i-1}^{(\mathrm{buf})}, \hat{\mathbf{Y}}_i\right], S_i\right), \tag{9}$$

where $S_i = \max(S_{i-1}, S_i)$ and $\mathsf{ZeroPadding}(\mathbf{A}, S)$ appends row zero vectors to $\mathbf{A}$ so that the first dimension becomes S. Next, the speaker permutation $\mathbf{P}_\psi$ for the current chunk is calculated between $\mathbf{Z}_{i-1}^{(\mathrm{buf})}$ and $\hat{\mathbf{Z}}_{i-1}^{(\mathrm{buf})}$ using Eq. (5). Then, the output for the current chunk is permuted as follows:

$$\mathbf{Y}_i = \mathbf{P}_\psi \hat{\mathbf{Z}}_i, \tag{10}$$

where $\mathbf{Y}_i$ is the final diarization result of the chunk i. After that, at most $L_{\max}$ time indexes $\mathcal{T} \subseteq \{1, \ldots, L_{i-1} + \Delta\}$ are selected based on the concatenated outputs $\left[\mathbf{Z}_{i-1}^{(\mathrm{buf})}, \mathbf{Y}_i\right] \in (0,1)^{S_i \times (L_{i-1}+T)}$, and the FLEX-STB is updated as follows:

$$\mathbf{X}_i^{(\mathrm{buf})} = \left[\mathbf{x}_\tau \mid \tau \in \mathcal{T}\right], \; \mathbf{Y}_i^{(\mathrm{buf})} = \left[\mathbf{y}_\tau \mid \tau \in \mathcal{T}\right], \tag{11}$$

where $\mathbf{x}_\tau$ is the τ-th column vector of $[\mathbf{X}_{i-1}^{(\mathrm{buf})}, \mathbf{X}_i]$, $\mathbf{y}_\tau$ is the τ-th column vector of $[\mathbf{Z}_{i-1}^{(\mathrm{buf})}, \mathbf{Y}_i]$. The frame selection strategies are described in Section 3.2.

3.2. Selection strategy

When the number of accumulated features becomes larger than the buffer size $L_{\max}$, a selection strategy is needed to keep relevant features that contain the speaker permutation information from $\left[\mathbf{X}_{i-1}^{(\mathrm{buf})}, \mathbf{X}_i\right]$ and $\left[\mathbf{Z}_{i-1}^{(\mathrm{buf})}, \mathbf{Y}_i\right]$. In this section, four selection functions are proposed for flexible numbers of speakers.

- **Uniform sampling**: Uniform distribution sampling is applied to extract $L_{\max}$ frames.
- **First-in-first-out (FIFO)**: The most recent $L_{\max}$ features and the corresponding diarization results are stored in the buffer, which follows the first-in-first-out manner.
- **Kullback-Leibler divergence based selection**: We utilize the Kullback-Leibler divergence (KLD) to measure the difference between two probability distributions: the speaker activities distribution and the uniform distribution at time t, which can be represented as follows:

$$\mathrm{KLD}_t = \sum_{s=1}^{S_i} p_{s,t} \log \frac{p_{s,t}}{q_{s,t}}, \tag{12}$$

$$p_{s,t} = \frac{r_{s,t}}{\sum_{s'=1}^{S_i} r_{s',t}}, \tag{13}$$

$$q_{s,t} = \frac{1}{S_i}, \tag{14}$$

  where $\left[\mathbf{Z}_{i-1}^{(\mathrm{buf})}, \mathbf{Y}_i\right] = (r_{s,t})_{\substack{1\le t\le(L_{i-1}+\Delta)\\ 1\le s\le S_i}}$ is the posteriors from EEND with FLEX-STB and $q_{s,t}$ is the uniform distribution. Top $L_{\max}$ samples with the highest KLD values are selected from $\left[\mathbf{Z}_{i-1}^{(\mathrm{buf})}, \mathbf{Y}_i\right]$ and the corresponding $\left[\mathbf{X}_{i-1}^{(\mathrm{buf})}, \mathbf{X}_i\right]$.
- **Weighted sampling using KLD**: The combination of uniform sampling and KLD based selection. $L_{\max}$ features are randomly selected with the probabilities which are proportional to KLD_t.

4. Experiment

4.1. Data

We generated 100k simulated mixtures of one to four speakers following the procedure in [30] using Switchboard-2 (Phase I, II, III), Switchboard Cellular (Part 1, 1), and the NIST Speaker Recognition Evaluation datasets (SRE). Additionally, we added noises from the MUSAN corpus [39] and room impulse responses (RIRs) from the Simulated Room Impulse Response Database [40]. These simulated mixtures were used for training the EEND-based model. Two real conversation datasets: the CALLHOME [12] and the DIHARD II [3] were prepared for evaluation.

4.2. Experiment setting

In this paper, we evaluated the proposed method on the offline EEND-EDA model. The EEND-EDA model was trained with four Transformer encoder blocks and 256 attention units containing four heads [31]. We firstly trained the model using a two-speaker dataset for 100 epochs and then finetuned with the concatenation of one- to four-speaker simulated datasets for 25 epochs. Finally, EEND-EDA model was finetuned using a development set of CALLHOME, or DIHARD II, respectively.

Table 2: *DERs (%) of online EEND-EDA with chunk size* $\Delta = 1\,\mathrm{s}$ *using FLEX-STB and offline EEND-EDA with chunk size* $\Delta = \infty$. *Note that all results are based on the estimated number of speakers, including the overlapping regions without oracle SAD.*

	Online ($\Delta = 1\,\mathrm{s}$)							
	CALLHOME			DIHARD II			**Offline** ($\Delta = \infty$)	
	$L_{\max} = 10\,\mathrm{s}$	$L_{\max} = 50\,\mathrm{s}$	$L_{\max} = 100\,\mathrm{s}$	$L_{\max} = 10\,\mathrm{s}$	$L_{\max} = 50\,\mathrm{s}$	$L_{\max} = 100\,\mathrm{s}$	CALLHOME	DIHARD II
FLEX-STB with selection strategy								
Uniform sampling	27.6	20.2	19.3	52.4	39.3	36.8	-	-
FIFO	29.5	**19.4**	**19.1**	57.2	41.1	37.0	-	-
KLD selection	30.0	22.3	20.9	52.6	40.8	37.7	-	-
Weighted sampling using KLD	**26.6**	20.0	19.5	**50.3**	**37.9**	**36.0**	-	-
Without FLEX-STB	-	-	-	-	-	-	**15.3**	**32.9**

We evaluated all systems with the diarization error rate (DER) metric in both overlapping and non-speech regions. A collar tolerance of 250 ms was applied at the start and end of each segment for the CALLHOME dataset. Following the regulation of the second DIHARD challenge [3], we did not use collar tolerance for the DIHARD II dataset.

4.3. Results

4.3.1. Effect of selection strategies and buffer size

Table 2 shows the effect of the selection strategies and the buffer size of the FLEX-STB on the EEND-EDA model in the left part. Experiment conditions varied from four selection methods with buffer sizes equal to $10\,\mathrm{s}$, $50\,\mathrm{s}$ and $100\,\mathrm{s}$ but fixed the chunk size Δ to $1\,\mathrm{s}$. All results were calculated with the estimated number of speakers including the *overlapping regions without oracle sound activity detection (SAD)*. It is shown that incremental buffer size which provides more input information improved the accuracy regardless of the selection strategies. Regarding the selection strategies, on most cases weighted sampling using KLD outperformed other strategies on both datasets. The best results from online system are $19.1\,\%$ and $36.0\,\%$ for CALLHOME and DIHARD II, respectively.

4.3.2. Comparison with the offline EEND-EDA system

We also compared the performance of our proposed online and baseline offline systems in Table 2. The input of the offline EEND-EDA system is the whole recording during inference while that for the online system is the $1\,\mathrm{s}$ chunk. Compared with the offline system, DERs of the online system increases by $3.8\,\%$ and $3.1\,\%$ on these two datasets, which would be acceptable degradation by considering the benefit of streaming diarization. The performance degradation is supposed to come from the mismatch between the offline model which was trained with fixed large chunk size and the online mechanism whose input sizes are incrementally increased.

4.3.3. Comparison with other online diarization systems

First, we compared our method with the recently proposed BW-EDA-EEND [35] on the CALLHOME dataset. Both methods were trained using the simulated data following the procedure in [30] with the same datasets. In order to compare with it in the same condition, we evaluated our method with a $10\,\mathrm{s}$ chunk size. As shown in Table 3, in a $10\,\mathrm{s}$ chunk size and estimated SAD condition, our proposed method outperforms the BW-EDA-EEND on all speaker-number cases.

Next, we compared our proposed method with other systems in more realistic scenario, i.e., DIHARD II. For a fair comparison with other online methods, we follow the DIHARD II track 1, where *the oracle SAD information is provided.* We used the oracle SAD information to filter out non-speech frames of the estimated diarization result. Table 4 shows the comparison with other systems. The proposed online EEND-EDA with FLEX-STB achieved a DER of $25.8\,\%$, which outperformed the UIS-RNN-SML, and is comparable to the *offline* DIHARD II baseline.

Table 3: *DERs (%) of each number of speakers on the CALLHOME dataset with* $10\,\mathrm{s}$ *chunk size. Both estimated the number of speakers and included the overlapping regions without using oracle SAD.*

	Number of speakers		
Method	2	3	4
BW-EDA-EEND [35]	11.8	18.3	26.0
EEND-EDA w/ FLEX-STB	**10.0**	**14.0**	**21.1**

Table 4: *DERs (%) on DIHARD II computed by using oracle SAD including overlapping regions. Online systems with STB were evaluated in a* $1\,\mathrm{s}$ *chunk size* Δ *and* $100\,\mathrm{s}$ *buffer size* $L_{\max}$.

Method	DER
DIHARD-2 baseline (offline) [3]	26.0
UIS-RNN-SML [24]	27.3
EEND-EDA w/ FLEX-STB	**25.8**

4.3.4. Real-time factor and latency

Our experiment was conducted on one NVIDIA Tesla P100 GPU. To calculate the average computing time of one buffer, we filled the buffer with dummy values for the first iteration to keep the buffer size always the same among chunks. The real-time factor was equal to 0.13 when we applied FLEX-STB to EEND-EDA with chunk size equal to $1\,\mathrm{s}$, and a buffer size of $100\,\mathrm{s}$. This means that the average computation duration of a $1\,\mathrm{s}$ chunk was $0.13\,\mathrm{s}$ which is acceptable for the online processing.

5. Conclusion

In this paper, we proposed an online streaming speaker diarization method that handles overlapping speech and flexible numbers of speakers. A speaker tracing buffer for flexible numbers of speakers was proposed to mitigate the different number of speakers among chunks. Experimental results showed that the proposed online system achieves comparable results with the offline method and better results than the BW-EDA-EEND online method. One of our future studies is to incorporate various extensions developed at the recent DIHARD III challenge, including semi-supervised training and model fusion [6].

6. References

[1] S. E. Tranter and D. A. Reynolds, "An overview of automatic speaker diarization systems," *IEEE Trans. on ASLP*, vol. 14, no. 5, pp. 1557–1565, 2006.

[2] X. Anguera, S. Bozonnet, N. W. D. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *IEEE Trans. on ASLP*, vol. 20, no. 2, pp. 356–370, 2012.

[3] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, "The second DIHARD diarization challenge: Dataset, task, and baselines," in *INTERSPEECH*, 2019, pp. 978–982.

[4] N. Ryant, K. Church, C. Cieri, J. Du, S. Ganapathy, and M. Liberman, "Third DIHARD challenge evaluation plan," *arXiv preprint arXiv:2006.05815*, 2020.

[5] T. J. Park, N. Kanda, D. Dimitriadis, K. J. Han, S. Watanabe, and S. Narayanan, "A review of speaker diarization: Recent advances with deep learning," *arXiv preprint arXiv:2101.09624*, 2021.

[6] S. Horiguchi, N. Yalta, P. Garcia, Y. Takashima, Y. Xue, D. Raj, Z. Huang, Y. Fujita, S. Watanabe, and S. Khudanpur, "The Hitachi-JHU DIHARD III system: Competitive end-to-end neural diarization and x-vector clustering systems combined by DOVER-lap," *arXiv preprint arXiv:2102.01363*, 2021.

[7] F. Landini, J. Profant, M. Diez, and L. Burget, "Bayesian hmm clustering of x-vector sequences (vbx) in speaker diarization: theory, implementation and analysis on standard tasks," 2020.

[8] W. Kang, B. C. Roy, and W. Chow, "Multimodal speaker diarization of real-world meetings using d-vectors with spatial features," in *ICASSP*, 2020, pp. 6509–6513.

[9] T. Yoshioka, I. Abramovski, C. Aksoylar, Z. Chen, M. David, D. Dimitriadis, Y. Gong, I. Gurvich, X. Huang, Y. Huang *et al.*, "Advances in online audio-visual meeting transcription," in *ASRU*, 2019, pp. 276–283.

[10] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal *et al.*, "The AMI meeting corpus: A pre-announcement," in *MLMI*, 2005, pp. 28–39.

[11] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke *et al.*, "The ICSI meeting corpus," in *ICASSP*, vol. 1, 2003, pp. I–I.

[12] "2000 NIST Speaker Recognition Evaluation," https://catalog.ldc.upenn.edu/LDC2001S97.

[13] A. Martin and M. Przybocki, "The NIST 1999 speaker recognition evaluation—an overview," *Digital signal processing*, vol. 10, no. 1-3, pp. 1–18, 2000.

[14] M. Senoussaoui, P. Kenny, T. Stafylakis, and P. Dumouchel, "A study of the cosine distance-based mean shift for telephone speech diarization," *IEEE Trans. on ASLP*, vol. 22, no. 1, pp. 217–227, 2013.

[15] J. Barker, S. Watanabe, E. Vincent, and J. Trmal, "The fifth "CHiME" speech separation and recognition challenge: Dataset, task and baselines," in *INTERSPEECH*, 2018.

[16] N. Kanda, R. Ikeshita, S. Horiguchi, Y. Fujita, K. Nagamatsu, X. Wang, V. Manohar, N. E. Y. Soplin, M. Maciejewski, S.-J. Chen *et al.*, "The Hitachi/JHU CHiME-5 system: Advances in speech recognition for everyday home environments using multiple microphone arrays," in *CHiME-5*, 2018.

[17] S. Watanabe, M. Mandel, J. Barker, and E. Vincent, "CHiME-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings," in *CHiME-6*, 2020.

[18] J. Geiger, F. Wallhoff, and G. Rigoll, "GMM-UBM based open-set online speaker diarization," in *INTERSPEECH*, 2010.

[19] K. Markov and S. Nakamura, "Improved novelty detection for online GMM based speaker diarization," in *INTERSPEECH*, 2008.

[20] S. Madikeri, I. Himawan, P. Motlicek, and M. Ferras, "Integrating online i-vector extractor with information bottleneck based speaker diarization system," in *INTERSPEECH*, 2015, pp. 3105–3109.

[21] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker diarization using deep neural network embeddings," in *ICASSP*, 2017, pp. 4930–4934.

[22] W. Zhu and J. Pelecanos, "Online speaker diarization using adapted i-vector transforms," in *ICASSP*, 2016, pp. 5045–5049.

[23] A. Zhang, Q. Wang, Z. Zhu, J. Paisley, and C. Wang, "Fully supervised speaker diarization," in *ICASSP*, 2019, pp. 6301–6305.

[24] E. Fini and A. Brutti, "Supervised online diarization with sample mean loss for multi-domain data," in *ICASSP*, 2020, pp. 7134–7138.

[25] M. Diez, L. Burget, S. Wang, J. Rohdin, and J. Černockỳ, "Bayesian HMM based x-vector clustering for speaker diarization," in *INTERSPEECH*, 2019, pp. 346–350.

[26] A. McCree, G. Sell, and D. Garcia-Romero, "Speaker diarization using leave-one-out gaussian PLDA clustering of dnn embeddings." in *INTERSPEECH*, 2019, pp. 381–385.

[27] Z. Huang, S. Watanabe, Y. Fujita, P. García, Y. Shao, D. Povey, and S. Khudanpur, "Speaker diarization with region proposal network," in *ICASSP*, 2020, pp. 6514–6518.

[28] Y. Fujita, N. Kanda, S. Horiguchi, K. Nagamatsu, and S. Watanabe, "End-to-end neural speaker diarization with permutation-free objectives," in *INTERSPEECH*, 2019, pp. 4300–4304.

[29] Y. Fujita, S. Watanabe, S. Horiguchi, Y. Xue, and K. Nagamatsu, "End-to-end neural diarization: Reformulating speaker diarization as simple multi-label classification," *arXiv preprint arXiv:2003.02966*, 2020.

[30] Y. Fujita, N. Kanda, S. Horiguchi, Y. Xue, K. Nagamatsu, and S. Watanabe, "End-to-end neural speaker diarization with self-attention," in *ASRU*, 2019, pp. 296–303.

[31] S. Horiguchi, Y. Fujita, S. Watanabe, Y. Xue, and K. Nagamatsu, "End-to-end speaker diarization for an unknown number of speakers with encoder-decoder based attractors," in *INTERSPEECH*, 2020, pp. 269–273.

[32] Y. Fujita, S. Watanabe, S. Horiguchi, Y. Xue, J. Shi, and K. Nagamatsu, "Neural speaker diarization with speaker-wise chain rule," *arXiv preprint arXiv:2006.01796*, 2020.

[33] T. von Neumann, K. Kinoshita, M. Delcroix, S. Araki, T. Nakatani, and R. Haeb-Umbach, "All-neural online source separation, counting, and diarization for meeting analysis," in *ICASSP*, 2019, pp. 91–95.

[34] K. Kinoshita, M. Delcroix, S. Araki, and T. Nakatani, "Tackling real noisy reverberant meetings with all-neural source separation, counting, and diarization system," in *ICASSP*, 2020, pp. 381–385.

[35] E. Han, C. Lee, and A. Stolcke, "BW-EDA-EEND: Streaming end-to-end neural speaker diarization for a variable number of speakers," *arXiv preprint arXiv:2011.02678*, 2020.

[36] Y. Takashima, Y. Fujita, S. Watanabe, S. Horiguchi, P. García, and K. Nagamatsu, "End-to-end speaker diarization conditioned on speech activity and overlap detection," in *SLT*, 2021, pp. 849–856.

[37] Y. Xue, S. Horiguchi, Y. Fujita, S. Watanabe, P. García, and K. Nagamatsu, "Online end-to-end neural diarization with speaker-tracing buffer," in *SLT*, 2021, pp. 841–848.

[38] Z. Dai, Z. Yang, Y. Yang, J. G. Carbonell, Q. Le, and R. Salakhutdinov, "Transformer-xl: Attentive language models beyond a fixed-length context," in *ACL*, 2019, pp. 2978–2988.

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

[40] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *ICASSP*, 2017, pp. 5220–5224.