# AUXILIARY LOSS OF TRANSFORMER WITH RESIDUAL CONNECTION FOR END-TO-END SPEAKER DIARIZATION

*Yechan Yu*[1§] *Dongkeon Park*[2§] *Hong Kook Kim*[1,2]

[1]School of Electrical Engineering and Computer Science, [2]AI Graduate School,
Gwangju Institute of Science and Technology, Gwangju 61005, Republic of Korea

yechan1202@gm.gist.ac.kr {dongkeon,hongkook}@gist.ac.kr

## ABSTRACT

End-to-end neural diarization (EEND) with self-attention directly predicts speaker labels from inputs and enables the handling of overlapped speech. Although the EEND outperforms clustering-based speaker diarization (SD), it cannot be further improved by simply increasing the number of encoder blocks because the last encoder block is dominantly supervised compared with lower blocks. This paper proposes a new residual auxiliary EEND (RX-EEND) learning architecture for transformers to enforce the lower encoder blocks to learn more accurately. The auxiliary loss is applied to the output of each encoder block, including the last encoder block. The effect of auxiliary loss on the learning of the encoder blocks can be further increased by adding a residual connection between the encoder blocks of the EEND. Performance evaluation and ablation study reveal that the auxiliary loss in the proposed RX-EEND provides relative reductions in the diarization error rate (DER) by 50.3% and 21.0% on the simulated and CALLHOME (CH) datasets, respectively, compared with self-attentive EEND (SA-EEND). Furthermore, the residual connection used in RX-EEND further relatively reduces the DER by 8.1% for CH dataset.

***Index Terms***— speaker diarization, end-to-end neural diarization, auxiliary loss, residual connection

## 1. INTRODUCTION

Speaker diarization (SD) is a process for identifying "who spoke when" by dividing an audio recording into homogeneous segments using speaker labels [1, 2, 3]. SD is essential to many speech-related applications with multi-speaker audio data, such as conversational multi-part speech recognition for business meetings or interviews and speaker-dependent video indexing [4, 5, 6].

In general, SD has been considered as a speaker clustering problem which assigns or classifies a speaker label to each speech segment. A clustering-based SD system typically has a modular structure, comprising speech activity detection, a speaker embedding extractor, and speaker clustering [7, 8, 9]. For a given utterance, each segment is represented by a speaker embedding vector, such as i-vectors [7, 8], d-vectors [10, 11], and x-vectors [12, 13]. After assigning a speaker label to each segment, all segments with the same speaker label are grouped into a cluster.

Although such clustering-based SD systems have performed reliably in many recent SD challenges [14, 15, 16], they mainly have two disadvantages. First, they have difficulty in handling speech segments in which more than two speakers overlap because the speaker embedding for each segment can be represented by only one speaker. Second, the performance of a clustering-based SD system is limited because all the modules are not jointly optimized.

End-to-end neural diarization (EEND) approaches have been proposed to overcome these disadvantages [17, 18]. Unlike traditional clustering-based methods [12, 13], EEND methods consider SD as multi-label classification; therefore, speech activity detection and overlapped speech detection modules are unnecessary in the EEND framework Consequently, the EEND method outperforms the clustering-based method for simulated and real datasets [17, 18]. Moreover, applying a self-attention mechanism to the EEND—self-attentive EEND (SA-EEND)—improves performance because self-attention could simultaneously accept global relation information over all frames [18].

Recently, several researchers have attempted to improve SA-EEND [19, 20]. Although SA-EEND provides adequate global attention over all frames, it is insufficient to deal with local information regarding speaker changes over several adjacent frames. The time-dilated convolutional network (TDCN), a sequential architecture of modeling local and global information, was applied to local embedding in SA-EEND [19]. In contrast, a conformer-based EEND (CB-EEND) was proposed to improve the performance of SA-EEND, where the transformer was replaced with the conformer in SA-EEND to capture local and global information simultaneously [20].

Although these approaches can improve performance, our preliminary experiment demonstrated that SA-EEND could be further improved by enhancing the learning strategy of the transformer. The preliminary experiment was motivated by previous studies establishing that increasing the number of encoder blocks in the transformer improved performance for automatic speech recognition [21] and natural language processing [22]. Accordingly, we increased the number of encoder blocks in SA-EEND from four to or more. Unfortunately, SD performance degraded when the number of encoder blocks was greater than four. Furthermore, the encoder blocks near the input layer contributed little to the performance compared with those close to the output layer, which will be explained in Sections 4.3 and 4.4. Therefore, the performance of SA-EEND should be improved when enabling the encoder blocks in the lower layer to learn more accurately for a greater contribution.

Based on this experiment, we propose a new residual auxiliary EEND (RX-EEND)-based learning architecture for transformers to enforce the lower encoder blocks to learn more accurately. There are two main contributions of this study: applying auxiliary loss and adding a residual connection. (1) When training the transformer in RX-EEND, the auxiliary loss is applied to the output of each encoder block, including the last encoder block. The additional auxiliary

[§]The first two authors contributed equally.

This research is supported by Ministry of Culture, Sports, and Tourism (MCST) and Korea Creative Content Agency (KOCCA) in the Culture Technology(CT) Research & Development Program(R2020060002) 2022, and by the GIST-MIT Research Collaboration grant funded by the GIST in 2022.

loss should strongly supervise each encoder block, as observed in [23]. Thus, the addition of encoder blocks improves performance, which will be discussed in Section 3.1. (2) The effect of auxiliary loss on the learning of the encoder blocks can be further increased by adding a residual connection between the encoder blocks of the EEND because residual connections enable gradients to flow directly across the encoder blocks. Furthermore, using the residual block provides ensemble models as described in [24]. Thus, the proposed RX-EEND should have a smaller generalization error than that of SA-EEND, which will be described in Section 3.2.

The rest of this paper is organized as follows. Section 2 briefly reviews conventional SA-EEND, and Section 3 proposes RX-EEND with auxiliary loss and residual connections. Next, Section 4 evaluates the performance of the proposed RX-EEND on the simulated and CALLHOME (CH) datasets. After that, an ablation study is conducted to demonstrate the individual contribution of auxiliary loss and residual connections each to the SD performance. In addition, the applicability of auxiliary loss and residual connections to conformer-EEND is discussed. Finally, Section 5 concludes the paper.

## 2. SELF-ATTENTIVE EEND

As described in the Introduction, the proposed RX-EEND is constructed based on the SA-EEND model [18] by adding a residual connection to each encoder block that is learned by its corresponding auxiliary loss. First, we briefly review conventional SA-EEND, as depicted in Fig. 1(a).

To extract input feature vectors, a 25-ms Hamming window with a hop size of 10-ms is applied to each utterance, and a 23-dimensional log-scaled mel-filterbank analysis is performed to each windowed speech frame. Next, each series of 15 frames is concatenated into a single dimensional feature vector, $\mathbf{x}_t \in \mathbb{R}^F$ with $F$ = 345, which is repeated with a stride of 10 frames, resulting in $X = [\mathbf{x}_1, \cdots, \mathbf{x}_T]$, where $T$ is the total number of feature vectors.

Next, $X$ is processed into a linear and layer normalization to obtain the input embedding vectors for the series of $P$ encoder blocks of EEND, $E^0 = [\mathbf{e}_1^0, \cdots, \mathbf{e}_T^0]$. Typically, $P$ = 4 in [18]. The $p$-th encoder block provides self-attentive features from the $(p-1)$-th embedding, as follows:

$$\mathbf{e}_t^0 = \mathrm{Norm}(\mathrm{Linear}^{\mathrm{F}}(\mathbf{x}_t)) \in \mathbb{R}^D, \tag{1}$$

$$E^p = \mathrm{Encoder}_p^{\mathrm{D}}(E^{p-1}),\ (1 \le p \le P). \tag{2}$$

After passing all the $P$ encoder blocks, the last output vectors, $E^P$, are applied to a linear and sigmoid function to get the posteriors $\hat{\mathbf{y}}_t = [\hat{y}_{t,1}, \cdots, \hat{y}_{t,S}]$ of $S$ speakers at time $t$, as follows:

$$\hat{\mathbf{y}}_t = \mathrm{sigmoid}(\mathrm{Linear}^{\mathrm{D}}(\mathbf{e}_t^P)). \tag{3}$$

In the training phase, SA-EEND is optimized using the permutation invariant scheme [17]. That is, the loss is calculated between $\hat{\mathbf{y}}_t$ and the ground truth labels $\mathbf{y}_t = [y_{t,1}, \cdots, y_{t,S}] \in \{0,1\}^S$, thus the total loss function, $\mathcal{L}_{\mathrm{d}}$ is defined as

$$\mathcal{L}_{\mathrm{d}} = \frac{1}{TS} \min_{\phi \in \mathrm{perm}(S)} \sum_{t=1}^{T} H\left(\mathbf{y}_t^{\phi}, \hat{\mathbf{y}}_t\right) \tag{4}$$

where $\mathrm{perm}(S)$ is the set of all possible permutations of speakers, $\mathbf{y}_t^{\phi} \in \{0,1\}^S$ is the set of the permuted ground truth labels according to $\phi$. In (4), $H(\mathbf{y}_t, \hat{\mathbf{y}}_t)$ is the binary cross-entropy defined as

$$H\left(\mathbf{y}_t, \hat{\mathbf{y}}_t\right) = \sum_s -y_{t,s} \log \hat{y}_{t,s} - (1 - y_{t,s}) \log\left(1 - \hat{y}_{t,s}\right). \tag{5}$$

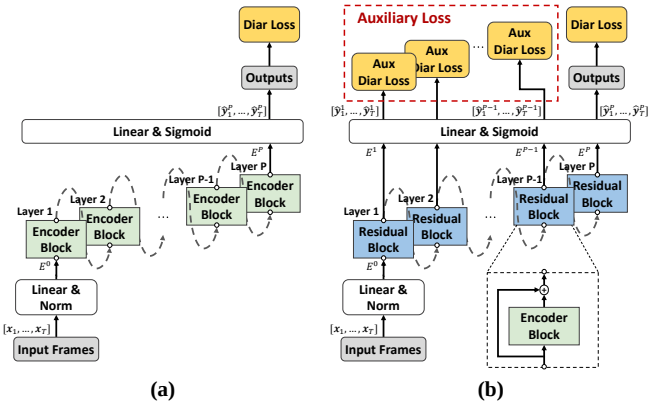


**Fig. 1**: *Network architectures of (a) conventional SA-EEND [18] and (b) proposed RX-EEND.*

## 3. PROPOSED METHOD

In the conventional SA-EEND, the single loss function in (4) is back-propagated into multiple number of encoder blocks sequentially. According to our preliminary experiment, the last encoder block contributes the most. Moreover, increasing the number of encoder blocks from four to or more degraded the performance. Thus, the RX-EEND is proposed to overcome this phenomenon by incorporating a new learning strategy and by adding a residual connection to each encoder block. The following subsections present a more detailed explanation for the proposed RX-EEND by comparing it with SA-EEND.

### 3.1. Auxiliary Loss

Inspired by the transformer-based object detection in [23], the first new approach for the proposed RX-EEND is to apply different auxiliary loss to each of encoder blocks. Accordingly, we apply a linear and sigmoid function to the outputs of each encoder block, whereas SA-EEND applies it only to the outputs of the last encoder block. As depicted in the upper part of Fig. 1(b), the posteriors estimated from the $p$-th encoder embedding vectors, $\hat{\mathbf{y}}_t^p$, are represented by

$$\hat{\mathbf{y}}_t^p = \mathrm{sigmoid}(\mathrm{Linear}^{\mathrm{D}}(\mathbf{e}_t^p))\ (1 \le p \le P-1). \tag{6}$$

Next, the loss function for training RX-EEND is defined as

$$\mathcal{L} = \mathcal{L}_{\mathrm{d}} + \lambda \mathcal{L}_{\mathrm{aux}} \tag{7}$$

where $\mathcal{L}_{\mathrm{d}}$ is the same to (4), and $\lambda$ is a hyperparameter to control the degree of effectiveness of the auxiliary loss against total loss. The auxiliary loss can be defined in two ways, depending on how the permutation of speakers is applied. The first version of $\mathcal{L}_{\mathrm{aux}}$ is defined as

$$\mathcal{L}_{\mathrm{aux}}^{\mathrm{shared}} = \frac{1}{TS(P-1)} \sum_{p=1}^{P-1} \sum_{t=1}^{T} H\left(\mathbf{y}_t^{\phi_P}, \hat{\mathbf{y}}_t^p\right) \tag{8}$$

where $\phi_P$ is the same permutation obtained from (4) and it is shared along all the lower encoder blocks. The second version of $\mathcal{L}_{\mathrm{aux}}$ optimizes $\phi_p$ individually for each encoder block, as follows:

$$\mathcal{L}_{\mathrm{aux}}^{\mathrm{indiv}} = \frac{1}{TS(P-1)} \sum_{p=1}^{P-1} \min_{\phi_p \in \mathrm{perm}(S)} \sum_{t=1}^{T} H\left(\mathbf{y}_t^{\phi_p}, \hat{\mathbf{y}}_t^p\right). \tag{9}$$

**Table 1**: *Distributions of simulated and real two speaker datasets for training and evaluating EENDs.*

| Data Style | | #Mixtures | Overlap ratio $\rho$ (%) |
|---|---|---|---|
| **Simulated Dataset** | | | |
| Sim2spk | Train | 100,000 | 34.1 |
| Sim2spk | Test | 500/500/500 | 34.4/27.3/19.6 |
| **Real Dataset** | | | |
| CALLHOME [25] | Train | 155 | 14.0 |
| CALLHOME [25] | Test | 148 | 13.1 |

Based on the performance comparison for different auxiliary loss functions between (8) and (9), which will be discussed in Section 4.4, $\mathcal{L}_{\text{aux}}^{\text{indiv}}$ in (9) is selected as the auxiliary loss of the proposed RX-EEND.

### 3.2. Residual Block

As the second approach for the performance improvement of EEND, the encoder block in SA-EEND is modified by adding a residual connection to increase the convergence speed by directly propagating gradients from the $p$-th encoder block to the $(p-1)$-th encoder block. Hereafter, we refer to an encoder block with a residual connection as a residual block, as depicted in the zoomed box in Fig. 1(b). Thus, the $p$-th residual block has the following function of

$$\mathbf{e}_t^p = \mathbf{e}_t^{p-1} + \text{Encoder}_p^{\text{D}}(\mathbf{e}_1^{p-1}, \cdots, \mathbf{e}_T^{p-1}) \; (1 \leq p \leq P). \quad (10)$$

The effectiveness of such residual blocks over encoder blocks will also be discussed in Sections 4.4 and 4.5.

## 4. EXPERIMENT

### 4.1. Datasets

We trained and evaluated the proposed RX-EEND by preparing simulated and real datasets, as depicted in Table 1. For the simulated dataset, denoted as Sim2spk, we first obtained utterances in a speaker-wise manner from Switchboard-2 (Phases I, II, and III), Switchboard Cellular (Parts 1 and 2), and the NIST Speaker Recognition Evaluation (2004, 2005, 2006, and 2008). Then, we randomly chose 10 to 20 utterances for each speaker. Next, the utterances were convolved with one of the simulated room impulse responses (RIRs) used in [26] and added noise signal from the MUSAN corpus [27]. The noisy utterances from one speaker were mixed with those from the other speaker according to the overlap ratio. This procedure was repeated so that the total number of mixture files was up to 100,000. For the detailed procedure on generating the simulated dataset, see [17]. For the real dataset, we used the telephone conversation dataset CH [25] (NIST SRE 2000; LDC2001S97, disk-8), which is the most widely used for SD studies. The CH dataset contained 500 sessions of multilingual telephonic speech. Each session had two to six speakers, and two dominant speakers were in each conversation. The distribution of the CH dataset described in Table 1 is identical to that in [18].

### 4.2. Experimental Setup

To examine the performance over the baselines in SA-EEND, the proposed RX-EEND was first designed to have four residual blocks, 256 attention units with four heads ($H$ = 4, $D$ = 256 and $P$ = 4 ), and 1,024 internal units in a position-wise feed-forward layer, which was identical to those in SA-EEND [18]. Then, the proposed RX-EEND and SA-EEND were trained using Sim2spk with $\rho$ = 34.1%, as shown in the 1st row of Table 1. On the other hand, the EENDs pretrained from Sim2spk dataset were finetuned for the real dataset. As a performance metric, a diarization error rate (DER) [28] with a collar tolerance of 0.25 s between the predicted outputs and targets was calculated. In particular, the DER for Sim2spk was measured once every overlap ratio. Throughout the experiments, we set $\lambda$ in (7) to 1 by the exhaustive search, and the Adam optimizer [29] was used, where the learning rate schedule with warm-up steps of 100,000 was applied [30]. And we set the number of training and adaptation epochs to be all 100 for simulated and real dataset, which setting was identical to that in [18].

**Table 2**: *Performance comparison in DER(%) between the proposed RX-EEND and SA-EEND.*

| | Sim2spk | | | Real |
|---|---|---|---|---|
| Method | $\rho$ = 34.4% | $\rho$ = 27.3% | $\rho$ = 19.6% | CH |
| SA-EEND [18] | 5.97 | 5.65 | 5.33 | 10.72 |
| RX-EEND | 4.18 | 3.93 | 4.01 | 9.17 |
| SA-EEND-deep | 10.33 | 10.30 | 9.56 | 12.62 |
| RX-EEND-deep | 3.13 | 2.84 | **2.63** | 7.69 |
| SA-EEND-large | 5.61 | 5.45 | 4.58 | 10.15 |
| RX-EEND-large | **2.74** | **2.45** | 2.72 | **7.37** |

### 4.3. Performance Comparison with SA-EEND

Table 2 compares the DERs of the proposed RX-EEND with those of SA-EEND on the simulated and real datasets. As shown in the first two rows of the table, RX-EEND outperformed SA-EEND for the two datasets. The lower DER for the real CH dataset implies that the residual connection of RX-EEND contributed adequately as a regularizer to reduce the generalization error.

Next, we investigated the effect of RX-EEND for deeper encoder or residual blocks by increasing the number of residual blocks of RX-EEND from four to eight. To this end, we set ($H$ = 4, $D$ = 256 and $P$ = 8) in both SA-EEND and the proposed RX-EEND, which were denoted as SA-EEND-deep and RX-EEND-deep, respectively. The 3rd and 4th rows of Table 2 compare the DERs between SA-EEND-deep and RX-EEND-deep. As pointed out in the Introduction, the DER of SA-EEND-deep with eight encoder blocks was higher than that of SA-EEND with four encoder blocks. However, the DERs of RX-EEND-deep were much lower for the two datasets than those of RX-EEND. Next, we observed the effectiveness of the auxiliary loss and residual connections for the different number of transformer heads and dimensions by changing $H$ = 4 and $D$ = 256 to $H$ = 8 and $D$ = 512 for both SA-EEND and RX-EEND, which were denoted as SA-EEND-large and RX-EEND-large in Table 2. As shown in the last two rows of Table 2, the DER reduction of SA-EEND-large was marginal compared with SA-EEND. In contrast, RX-EEND-large reduced DERs for the real dataset and Sim2spk with overlap ratios of $\rho$ = 34.4% and $\rho$ = 27.3%.

### 4.4. Analysis of the Contribution of Each Block

We examined the contribution of each encoder or residual block in SA-EEND or RX-EEND, respectively, by obtaining the embedding vectors at each block, $E^p$ for $p = 1, \cdots, P$. Fig. 2 illustrates the T-SNE plots of $E^4$ and $E^8$ for SA-EEND-large and RX-EEND-large, respectively, applied to one mixture from Sim2spk test data. It was

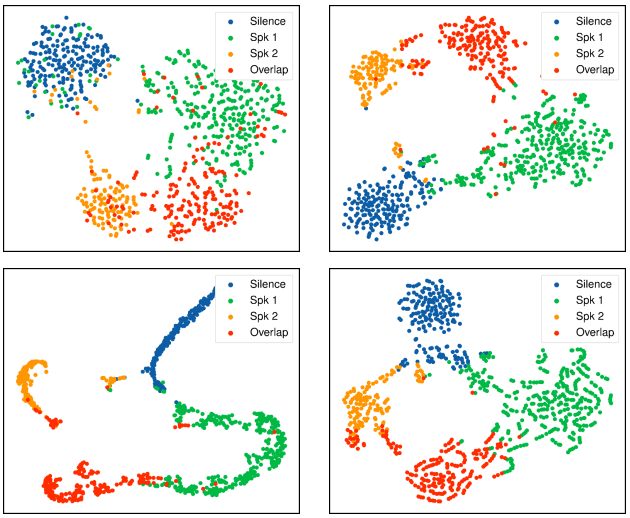


**Fig. 2**: *T-SNE plots of embedding vectors extracted from the fourth and eighth blocks of (a) SA-EEND-large and (b) proposed RX-EEND-large applied to one mixture from Sim2spk with $\rho$ = 34.4%.*

**Table 3**: *Comparison of DERs (%) of SA-EEND and RX-EEND using the embedding vectors at each block.*

| Block number | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| SA-EEND-large | 82.25 | 79.04 | 64.70 | 71.75 | 60.02 | 40.86 | 45.00 | 5.61 |
| RX-EEND-large | 27.27 | 13.46 | 7.28 | 4.95 | 4.06 | 3.16 | 2.89 | **2.74** |

shown from the plots for $E^8$ that SA-EEND-large and RX-EEND-large distinctly separated silence and each speaker because $E^8$ was related to the actual output for the two models. However, SA-EEND-large failed to separate silence and speakers when $E^4$ was used. In contrast, the proposed RX-EEND-large could reasonably separate silence and speaker even with the lower block, $E^4$.

Next, we predicted $\hat{\mathbf{y}}_t$ in (6) using Sim2spk test data with $\rho$ = 34.4% for SA-EEND-large and RX-EEND-large, respectively. Table 3 compares the DERs for each block of SA-EEND-large and RX-EEND-large. The DER tended to decrease as the number of blocks increased for both models. However, except for the last block, the DERs at all the lower blocks for SA-EEND-large were extremely high, so that the performance in SA-EEND seemed to be highly dependent on the last block. Instead, the proposed RX-EEND-large could supervise each residual block to learn through the auxiliary loss and residual connection for a greater contribution. Consequently, these results support our motivation for proposing an auxiliary loss and a residual block.

### 4.5. Ablation Studies

Ablation studies were conducted to evaluate the effect of the auxiliary loss and residual blocks of the proposed RX-EEND on performance. Table 4 compares DERs on Sim2spk test data with $\rho$ = 34.4% and CH dataset according to different variants of the proposed RX-EEND. Note here that SA-EEND-large and RX-EEND-large were used for these studies, and the 1st row of the table corresponds to the DERs for SA-EEND-large, which was identical to the DERs in Table 2.

First, we only replaced encoder blocks of SA-EEND-large with residual blocks. As shown in the 2nd row of the Table 4, we achieved the lowered DERs by just adding residual connections to the encoder blocks, while the DER reduction was not significant. Next, we examined the performance difference between two different auxiliary loss types described in (8) and (9). We applied each auxiliary loss type to SA-EEND-large (i.e., we did not use residual connections to the encoder blocks of SA-EEND). As shown in the 3rd and 4th rows of the table, the two types reduced DERs, and the auxiliary type of (9) was superior to that of (8). In addition, the residual connections were more effective than auxiliary loss for the CH dataset, as depicted in the last row.

**Table 4**: *Ablation study of residual connections and auxiliary loss used for RX-EEND, measured in DER (%).*

| Residual | $\mathcal{L}_{\text{aux}}^{\text{shared}}$ | $\mathcal{L}_{\text{aux}}^{\text{indiv}}$ | Sim2spk | Real |
|---|---|---|---|---|
| | | | 5.61 | 10.15 |
| ✓ | | | 5.58 | 10.00 |
| | ✓ | | 5.58 | 9.91 |
| | | ✓ | 2.79 | 8.02 |
| ✓ | | ✓ | **2.74** | **7.37** |

**Table 5**: *Comparison of DERs (%) when residual connections and auxiliary loss were applied to conformer-based EEND.*

| | Sim2spk | | | Real |
|---|---|---|---|---|
| Method | $\rho$ = 34.4% | $\rho$ = 27.3% | $\rho$ = 19.6% | CH |
| CB-EEND [20] | 2.85 | N/A | N/A | 9.70 |
| Conformer-EEND | 3.90 | 3.67 | 3.89 | 9.74 |
| RX-Conformer-EEND | **2.79** | **2.54** | **2.34** | **8.31** |

Finally, we examined whether the residual connections and auxiliary loss could be applied to other forms of EENDs. Accordingly, CB-EEND [20] was chosen as a baseline of this study. For a fair comparison with SA-EEND and RX-EEND, SpecAugment and convolutional subsampling in CB-EEND were removed, which was referred to as Conformer-EEND in Table 5. We then constructed RX-conformer-EEND by adding residual connections to conformer blocks and training with auxiliary loss. As shown in Table 5, the DERs of Conformer-EEND were higher than those of CB-EEND, implying that the subsampled features in CB-EEND could be dominant in DER reduction. However, applying our proposed approaches to Conformer-EEND overcame this performance degradation; even the RX-conformer-EEND was significantly superior to CB-EEND in real dataset. Finally, this study revealed that the residual connections and auxiliary loss in RX-EEND could also improve performance for other forms of EENDs.

## 5. CONCLUSIONS

In this paper, we proposed RX-EEND using auxiliary loss and residual connections to SA-EEND. The auxiliary loss supervised each encoder block individually to perform the same objective using residual connections in each encoder block to reduce the EEND generalization error. The proposed RX-EEND performance was evaluated on simulated and real datasets and compared with that of SA-EEND. The experimental results revealed that RX-EEND achieved significantly lower DERs than SA-EEND. Furthermore, based on the ablation studies, the residual connections or auxiliary loss contributed to reducing DERs and could be applied to other forms of EEND, such as CB-EEND. As future research, we will apply RX-EEND to a dataset with three or more speakers in future research.

## 6. REFERENCES


[1] S. E. Tranter, K. Yu, D. A. Reynolds, G. Evermann, D. Y. Kim, and P. C. Woodland, "An investigation into the the interactions between speaker diarisation systems and automatic speech transcription," *CUED/F-INFENG/TR-464*, 2003.

[2] S. E. Tranter and D. A. Reynolds, "An overview of automatic speaker diarization systems," *IEEE/ACM TASLP*, vol. 14, no. 5, pp. 1557–1565, 2006.

[3] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *IEEE/ACM TASLP*, vol. 20, no. 2, pp. 356–370, Feb 2012.

[4] D. A. Reynolds and P. Torres-Carrasquillo, "Approaches and applications of audio diarization," in *Proc. ICASSP*, 2005, pp. 953–956.

[5] S. E. Tranter and D. Reynolds, "Speaker diarisation for broadcast news," in *Proc. Odyssey*, 2004, pp. 337–344.

[6] AMI, "AMI Consortium," `http://www.amiproject.org/index.html`.

[7] S. H. Shum, N. Dehak, R. Dehak, and J. R. Glass, "Unsupervised methods for speaker diarization: An integrated and iterative approach," *IEEE/ACM TASLP*, vol. 21, no. 10, pp. 2015–2028, 2013.

[8] G. Sell and D. Garcia-Romero, "Speaker diarization with PLDA i-vector scoring and unsupervised calibration," in *Proc. SLT*, 2014, pp. 413–417.

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

[10] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *Proc. ICASSP*, 2018, pp. 4879–4883.

[11] Q. Wang, C. Downey, L. Wan, P. A. Mansfield, and I. L. Moreno, "Speaker diarization with LSTM," in *Proc. ICASSP*, 2018, pp. 5239–5243.

[12] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. ICASSP*, 2018, pp. 5329–5333.

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

[14] M. Diez, F. Landini, L. Burget, J. Rohdin, A. Silnova, K. Zmolikova, O. Novotný, K. Veselý, O. Glembek, O. Plchot, L. Mosner, and P. Matejka, "BUT system for DIHARD speech diarization challenge 2018," in *Proc. Interspeech*, 2018, pp. 2798–2802.

[15] 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 *Proc. Interspeech*, 2019, pp. 978–982.

[16] W. Wang, D. Cai, Q. Lin, L. Yang, J. Wang, J. Wang, and M. Li, "The dku-dukeece-lenovo system for the diarization task of the 2021 voxceleb speaker recognition challenge," *arXiv preprint arXiv:2109.02002*, 2021.

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

[18] F. Yusuke, K. Naoyuki, H. Shota, X. Yawen, N. Kenji, and W. Shinji, "End-to-end neural speaker diarization with self-attention," in *Proc. ASRU*, 2019, pp. 296–303.

[19] S. Maiti, H. Erdogan, K. Wilson, S. Wisdom, S. Watanabe, and J. R. Hershey, "End-to-end diarization for variable number of speakers with local-global networks and discriminative speaker embeddings," in *Proc. ICASSP*, 2021, pp. 7183–7187.

[20] Y. C. Liu, E. Han, C. Lee, and A. Stolcke, "End-to-end neural diarization: From transformer to conformer," in *Proc. Interspeech*, 2021, pp. 3081–3085.

[21] L. Dong, S. Xu, and B. Xu, "Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition," in *Proc. ICASSP*, 2018, pp. 5884–5888.

[22] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL*, 2019, pp. 4171–4186.

[23] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. ECCV*, 2020, pp. 213–229.

[24] A. Veit, M. J. Wilber, and S. Belongie, "Residual networks behave like ensembles of relatively shallow networks," in *Proc. NIPS*, 2016, pp. 550–558.

[25] M. Przybocki and A. Martin, *2000 NIST Speaker Recognition Evaluation (LDC2001S97)*, Philadelphia, New Jersey: Linguistic Data Consortium, 2001.

[26] K. Tom, P. Vijayaditya, P. Daniel, S. Michael L, and K. Sanjeev, "A study on data augmentation of reverberant speech for robust speech recognition," in *Proc. ICASSP*, 2017, pp. 5220–5224.

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

[28] J. G. Fiscus, J. Ajot, M. Michel, and J. S. Garofolo, "The rich transcription 2006 spring meeting recognition evaluation," in *Proc. MLMI*, 2006, pp. 309–322.

[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.

[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NIPS*, 2017, pp. 5998–6008.