INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Self-Distillation into Self-Attention Heads for Improving Transformer-based End-to-End Neural Speaker Diarization

Ye-Rin Jeoung, Jeong-Hwan Choi, Ju-Seok Seong, JeHyun Kyung, Joon-Hyuk Chang

Department of Electronic Engineering, Hanyang University, Seoul, Republic of Korea
{jyr0328, brent1104, jehyunkyung, as2835510, jchang}@hanyang.ac.kr

Abstract

In this study, we explore self-distillation (SD) techniques to improve the performance of the transformer-encoder-based self-attentive (SA) end-to-end neural speaker diarization (EEND). We first apply the SD approaches, introduced in the automatic speech recognition field, to the SA-EEND model to confirm their potential for speaker diarization. Then, we propose two novel SD methods for the SA-EEND, which distill the prediction output of the model or the SA heads of the upper blocks into the SA heads of the lower blocks. Consequently, we expect the high-level speaker-discriminative knowledge learned by the upper blocks to be shared across the lower blocks, thereby enabling the SA heads of the lower blocks to effectively capture the discriminative patterns of overlapped speech of multiple speakers. Experimental results on the simulated and CALLHOME datasets show that the SD generally improves the baseline performance, and the proposed methods outperform the conventional SD approaches.

Index Terms: speaker diarization, end-to-end neural diarization, self-attention mechanism, fine-tuning, self-distillation

1. Introduction

Speaker diarization task determines “who spoke when” from an audio mixture of multiple overlapping utterances acquired in a conversational environment. Conventionally, speaker diarization systems were implemented using several modules with different functions, for example, a voice activity detector, speaker embedding extractor, and clustering, to assign the speaker labels to the audio segments. However, such modular approaches are limited because they can only assign one speaker per segment, which is not suitable for handling overlapped speech, and cannot be directly optimized to reduce diarization errors. To overcome these limitations, an end-to-end neural speaker diarization (EEND) approaches were proposed [1, 2], which directly predicts the speaker activity labels from the audio mixture. Specifically, Fujita *et al.* [2] trained the transformer-based self-attentive (SA) EEND model using the binary cross-entropy (BCE) loss based on permutation invariant training (PIT) and achieved significant performance improvements.

Recently, researchers have been interested in effectively training the SA-EEND by assigning auxiliary losses [3, 4]. Yu *et al.* [3] demonstrated that the lower blocks of the SA-EEND model contribute less to diarization performance compared to higher blocks as the depth of the model increases. Based on these observations, they proposed residual auxiliary EEND (RX-EEND), which adds residual connections between blocks and computes the diarization loss for every block so that all blocks can be involved to the calculation of the loss function. Jeoung *et al.* [4] showed that the attention weight matrices of multi-head self-attention (MHSA), learn redundant patterns. To alleviate this issue, they proposed new auxiliary losses that leverage overlapped and speaker-wise speech activity patterns to guide the SA heads.

In this study, we propose a novel self-distillation (SD) method to effectively train the SA heads of the SA-EEND model based on prior research, which showed that the contribution of each layer to the diarization performance is different and that auxiliary losses for SA heads are effective. The main concept of SD is to divide a single network into the upper and lower layers based on their depth and then distill the higher-level knowledge learned from the upper layers into the lower layers [5]; SD was shown to be effective for training deep learning models in various fields [5–10]. First, we explore the applicability of SD techniques for speaker diarization tasks by adopting the algorithm proposed in the field of automatic speech recognition [6]. Subsequently, we propose new SD techniques, which distill high-level speaker-discriminative knowledge to the attention weight matrices of lower blocks. Our proposed technique can be implemented twofold: (1) output-to-head SD utilizing the speaker posteriors predicted from the output layer and (2) heads-to-head SD using the attention weight matrices extracted from the upper blocks. We expect that our methods help the attention weight matrices of lower blocks to contribute more to the prediction of speaker activity labels. The experiment on both simulation and real datasets for two speakers verified the effectiveness of our approaches.

2. Related work

2.1. Self-attentive end-to-end neural diarization

We briefly summarize the SA-EEND [2] as the baseline model. To predict the sequence of speaker activity labels, the SA-EEND model converts the T-length input feature $X = [\mathbf{x}_1, \cdots, \mathbf{x}_T]$ into an embedding $\mathbf{e}_t^0$ by passing it through a linear layer.

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

where t is the time frame index. Then, the blocks of the stacked transformer encoder are introduced.

$$E^p = \text{EncoderBlock}_p(E^{p-1}), 1 \leq p \leq P, \tag{2}$$

where $E^p = [\mathbf{e}_1^p, \cdots, \mathbf{e}_T^p]$ denotes the embedding sequence of the p-th block. The encoder block comprises an MHSA and feed-forward network, each of which is preceded by a layer normalization (LN) [11] and uses a residual connection. The feed-forward network consists of two linear layers and a ReLU activation [12]. The SA head in MHSA calculates using the scaled dot-product attention [13] as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(QK^\top/\sqrt{d}\right)V = HV, \tag{3}$$

10.21437/Interspeech.2023-1404

where $H \in \mathbb{R}^{T\times T}$ is the attention weight matrix that considers global feature relations, which is defined as the product of query Q and key K divided by the squared root of d. $V \in \mathbb{R}^{T\times d}$ denote the value and d is the dimension of hidden space, respectively. After passing through all the P encoder blocks, the prediction for S speakers, $\hat{\mathbf{y}}_t = [\hat{y}_{t,1}, \cdots, \hat{y}_{t,S}]$, at the time frame t is obtained as follows:

$$\hat{y}_t = \text{sigmoid}(o_t), \tag{4}$$

$$o_t = \text{Linear}\left(\text{LN}(\mathbf{e}_t^P)\right). \tag{5}$$

In the training step, the SA-EEND is optimized with a loss function $\mathcal{L}_d$, which is calculated between prediction and target label.

$$\mathcal{L}_d = \frac{1}{TS} \min_{\phi_1,\cdots,\phi_S \in \Phi_S} \sum_{t=1}^{T} \sum_{s=1}^{S} \text{BCE}(y_{t,s}^{\phi_s}, \hat{y}_{t,s}), \tag{6}$$

where Φ_S represents all possible permutations of the speakers, and $\mathbf{y}_{\phi_s} = [y_{1,s}^{\phi_s}, \cdots, y_{T,s}^{\phi_s}] \in \{0,1\}^T$ is the speaker label sequence according to the permutation ϕ_s, respectively.

2.2. Self-distillation

Knowledge distillation (KD) technique is a commonly used method for compressing models using a teacher network to distill knowledge into a student network [14]. Recently, several problems have been noted with KD: the quality and content of the knowledge that a student network learns depending on how the teacher network is designed, and student networks do not fully utilize knowledge [5]. The SD framework has been introduced in various fields [5–7] to overcome the limitations of KD, involving distilling knowledge within a single network, resulting in reduced training time and improved distillation accuracy. Furthermore, Xu *et al.* [6] proposed neighboring feature SD (NFSD) and attention-based feature SD (AFSD) methods for the speech recognition task, which distill the output information between the transformer encoder–decoder blocks.

This subsection introduces the application of the NFSD and AFSD for SA-EEND. The NFSD distills the output of the upper encoder block to the lower block within each group, considering two adjacent encoder blocks as a group. The loss function of NFSD is calculated using mean squared error (MSE) as follows:

$$\mathcal{L}_{NFSD} = \sum_{p=1}^{P/2} \text{MSE}(E_{2p-1}, E_{2p}), \tag{7}$$

The AFSD calculates dot-attention to integrate information obtained from the outputs of all upper blocks and distill it into lower blocks. The definition of the knowledge that the output feature of the p-th block needs to learn is denoted as G_p.

$$G_p = \sum_{l=p+1}^{P} \frac{\exp\left(\text{DotAttention}(E_p, E_l)\right)}{\sum_{l'=p+1}^{P} \exp\left(\text{DotAttention}(E_p, E_{l'})\right)} \odot E_l, \tag{8}$$

where $\odot$ indicates the Hadamard product. G_p represents the summation along the weighted outputs of the upper blocks. Then, AFSD loss is also calculated using the MSE as follows:

$$\mathcal{L}_{AFSD} = \sum_{p=1}^{P-1} \text{MSE}(E_p, G_p). \tag{9}$$

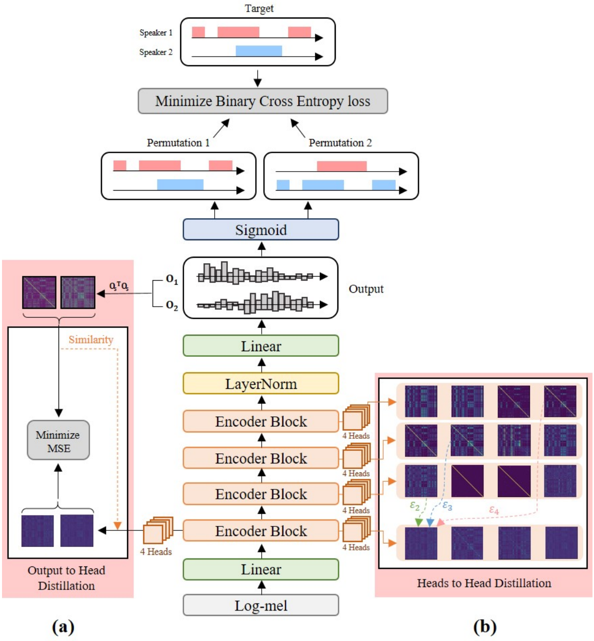

Figure 1: *SA-EEND model with two proposed SD methods. (a) Output-to-head distillation, (b) Heads-to-head distillation*

3. Proposed self-distillation method

We propose two methods to self-distill the knowledge into SA heads. We focus on the observation of previous studies that SA heads of the encoder block learn patterns relevant to speech and speaker information [2, 4]. Concurrently, we assume that the prediction outputs and SA heads of the higher blocks, which are closer to the output layer of the model, contain more information relevant to diarization prediction. Thus, we expect that the two proposed methods help each of the SA head in lower blocks learn patterns explicitly and achieve diarization performance improvement.

3.1. Output-to-head distillation

As the source information to be distilled into the SA heads of the lower blocks, we extract the output feature vector, $\mathbf{O}_s = [o_{1,s}, \cdots, o_{T,s}]$, containing the highest-level speaker-discriminative information. The target matrices to be distilled are generated as $\mathcal{M}_s = \mathbf{O}_s^\top \mathbf{O}_s (1 \leq s \leq S)$, which are used for computing the similarity and the training loss. To determine the SA head to which the knowledge is distilled, we measure the MSE between H_p^m, the attention weight matrix of the p-th encoder block, and $\mathcal{M}_s$. We select the SA head with the lowest similarity in the attention weight matrices to $\mathcal{M}_s$. Subsequently, we constrain the SA head to be similar to $\mathcal{M}_s$ using the loss function described below:

$$\mathcal{L}_{O2H} = \sum_{s=1}^{S} \max_{m\in M} \text{MSE}(H_p^m, \mathcal{M}_s), \tag{10}$$

where M is the number of attention heads of p-th block. We denote this SD method as *Output-to-Head* distillation. The overall process of the *Output-to-Head* distillation is in Figure 1(a).

3.2. Heads-to-head distillation

The high-level information from upper blocks is not fully used in the *Output-to-Head* distillation. Inspired by AFSD, we utilize the SA heads in upper blocks to distill the information into the SA heads of the lower block. We search for the suitable target $\hat{H}_k^{m'}$, m'-th attention weight matrix of p-th encoder block to be distilled into H_p^m, where p is smaller than k. We choose $\hat{H}_k^{m'}$, which has the maximum value of MSE, as H_p^m, similar to the *Output-to-Head* method. The information of the SA heads in p-th block to be distilled into SA heads in k-th block is defined as follows:

$$\mathcal{A}_k = \sum_{m=1}^{M} \max_{m' \in M} \mathrm{MSE}(H_p^m, \hat{H}_k^{m'}). \tag{11}$$

Furthermore, we apply the attention mechanism to design weights based on the amount of information upper SA heads indicate to share. The attention value ε_k, when the SA heads of p-th encoder block are distilled from the SA heads of k-th encoder block, is defined as follows:

$$\varepsilon_k = \frac{\mathcal{A}_k}{\sum_{k=p+1}^{P} \mathcal{A}_k}. \tag{12}$$

Figure 1(b) shows an example in which the p is set to one. It showed that the first head of the first block is distilled from the upper blocks. The loss function with attention applied for each layer is calculated as:

$$\mathcal{L}_{H2H} = \sum_{k=p+1}^{P} \varepsilon_k \cdot \mathcal{A}_k. \tag{13}$$

We define this SD method as *Heads-to-Head* distillation.

4. Experiments

The details of our experiments are introduced in this section. We configured our computing infrastructure with a single NVIDIA GeForce RTX 3090 GPU and conducted experiments using PyTorch version 1.10.1 on Ubuntu 18.04. To implement various approaches on SA-EEND, we modified public code on `https://github.com/hitachi-speech/EEND`.

4.1. Datasets

We prepared simulated and real datasets as same as described in [2]. The speech mixture simulation algorithm [2] was used to generate the simulated dataset, denoted as Sim2spk. The Switchboard-2 (Phases I, II, and III), Switchboard Cellular (Parts 1 and 2) [15], and NIST Speaker Recognition Assessment (2004, 2005, 2006, and 2008) corpora [16–19] were used to generate Sim2spk. All of these corpora were sampled at 8 kHz. Sim2spk used two different speakers for every utterance, generated by randomly selecting 10 to 20 utterances from the other. Each utterance was added with background noise samples from the MUSAN [20] and was convolved with a randomly chosen simulated room impulse response with a probability of 0.5, as described in [2]. A set of two-speaker telephone conversation utterances from the CALLHOME (CH) [21] dataset was used for the real evaluation dataset. Following [2], two speaker recordings from the CH dataset were split into adaptation and test set. The details of these datasets are shown in Table 1 including the number of mixtures and overlap ratios.

4.2. Experimental setup

The SA-EEND [2] with four transformer encoder blocks, each using four heads, was used as the baseline model. The number of model parameters was 5.35M. The input features were 23-dimensional log-scaled mel-filterbank energies with a 25 ms frame length and 10 ms frame shift. The training loss was calculated as the sum of $\mathcal{L}_d$ and an auxiliary loss function multiplied by a scaling factor. For $\mathcal{L}_{NFSD}$ and $\mathcal{L}_{AFSD}$, the same hyperparameters described in [6] were employed, with the scaling factors set to 1.0. Because the SA-EEND consisted of four encoder blocks, $\mathcal{L}_{NFSD}$ was applied to two groups: one comprising the first two encoder blocks and the other comprising the last two. In addition, $\mathcal{L}_{O2H}$ and $\mathcal{L}_{H2H}$ were applied using the scaling factors of 1 and 0.2, respectively. Because the datasets consisted of two speakers, $\mathcal{L}_{O2H}$ was applied to two SA heads selected. The Adam optimizer [22] was applied with 100,000 warmup steps in all training stages, and the learning rate was set to 0.00001 only during adaptation. The model was pretrained and fine-tuned for 100 epochs each using the Sim2spk dataset, and the CH two-speaker dataset was used for the adaptation step for additional 100 epochs. After the pretraining was completed, the parameters of the models obtained from the last 10 epochs were averaged and subsequently used for fine-tuning or adaptation. The evaluation was also conducted using the model subjected to the aforementioned parameter averaging scheme. The diarization error rate (DER) [23] was used as the evaluation metric, with the collar tolerance of 0.25 s for both the start and end of each segment. We applied an 11-frame median filter and a threshold of 0.5 for the final speaker activity prediction.

Table 1: *Statistics of simulated and real datasets*

Datasets	Sim2spk		CALLHOME	
	Train	Test	Adapt	Test
# mixtures	100,000	500 / 500 / 500	155	148
overlap ratio (%)	34.4	34.4 / 27.3 / 19.6	14.0	13.1

Table 2: *DERs (%) of SA-EEND by applying different auxiliary losses on two training steps. Pretraining (✗), Fine-tuning(✓)*

Auxiliary loss	Step	Sim2spk			Real
		34.4%	27.3%	19.6%	CH
- [2]	✗	6.30	6.14	6.17	10.46
$\mathcal{L}_{\text{aux}}$ [3]	✗	4.28	3.76	3.86	9.04
$\mathcal{L}_S$, $\mathcal{L}_O$ [4]	✗	4.29	4.11	4.15	8.67
$\mathcal{L}_{NFSD}$ [6]	✓	4.24	3.87	3.98	9.06
$\mathcal{L}_{AFSD}$ [6]	✓	4.04	3.89	3.88	9.01
$\mathcal{L}_{O2H}$	✗	5.13	5.14	5.37	9.26
	✓	4.07	3.75	3.72	8.58
$\mathcal{L}_{H2H}$	✗	4.75	4.41	4.41	9.11
	✓	**3.97**	**3.66**	**3.47**	**8.53**

5. Results and Analysis

5.1. Performance Comparison

Table 2 shows the performance of EEND models trained using various auxiliary losses during the pretraining and fine-tuning steps. We implemented the SA-EEND [2], RX-EEND [3], and the SA-EEND model with the auxiliary losses described in [4]; for the last one, the best experimental setup explored in [4] was adopted. $\mathcal{L}_{\text{aux}}$ [3] is an auxiliary loss computed using the same way as $\mathcal{L}_d$ [2] in all encoder blocks except the last encoder block. In addition, $\mathcal{L}_S$ and $\mathcal{L}_O$ [4] are auxiliary losses computed using BCE and MSE between specific SA heads and the matrices generated from two-speaker activity target, and between a specific SA head and overall speech activity pattern, respectively. The experimental results for two proposed losses in Table 2 were obtained when the SD was applied to the first en-

Table 3: *DERs (%) on the Sim2spk depending on the location of the encoder block where information is distilled.*

Loss	**Distillated encoder**				**Sim2spk**		
	1st	2nd	3rd	4th	34.4 %	27.3 %	19.6 %
$\mathcal{L}_{O2H}$	✓				**4.07**	**3.75**	**3.72**
	✓	✓			4.07	3.81	4.11
	✓	✓	✓		4.12	4.08	4.13
	✓	✓	✓	✓	4.14	4.10	4.11
$\mathcal{L}_{H2H}$	✓				**3.97**	**3.66**	**3.47**
	✓	✓			4.25	3.88	3.88
	✓	✓	✓		4.16	3.77	3.61

coder block. On both the simulated and real datasets, these proposed SD methods achieved lower DER than NFSD and AFSD that used the output of each encoder block for SD. Because the auxiliary losses were only applied during the pretraining step in [3, 4], we also conducted experiments by applying the proposed SD methods only during the pretraining step. As shown in Table 2, the application of the SD techniques during the fine-tuning step significantly improved the baseline system on both datasets, producing competitive performances with other models trained using the auxiliary losses proposd in [3,4]. However, when the SD methods were only applied during the pretraining step, the performance improvement was marginal. This could be because it is generally accepted that the SD method is best applied after the model has been fitted [6]. Thus, it can be explained that the proposed SD methods were also more effective when applied after the model has been sufficiently trained in advance. For the remainder of the paper, we applied the proposed SD methods during the fine-tuning stage.

5.2. Ablation study on selection of distilled encoder

Table 3 investigates the effect of the proposed SD losses on the different locations of the encoder blocks subjected to SD. Because SD aims to share high-level information across the lower layers of the model, the experiments were conducted by gradually expanding the number of encoder blocks, subjected to the proposed SD methods, from the lowest to the highest layers. Interestingly, exclusively applying the proposed loss functions to the first encoder block yielded superior performance than applying them to all lower blocks.

5.3. Effects of SD on SA-EEND

We visualized the four attention weight matrices of the encoder blocks of the trained models in Figure 2 to investigate the impact of the SD losses on SA heads. In the figure, the first three rows represent the attention weight matrices calculated in the first encoder block, and the fourth and fifth rows correspond to the second encoder block. Comparing Figure 2(a) and (b), we confirmed that the four attention heads show more explicit patterns with a larger weight by applying $\mathcal{L}_{O2H}$. In the case of $\mathcal{L}_{H2H}$ represented in Figure 2(c), the pattern of identity term with small weight was learned in head1 and head4, and the pattern of head3 was more apparent than 2(b). Thus, considering the DER in Table 2 and Figure 2, it can be inferred that our SD losses helped the SA heads of the lower block to learn the speech pattern effectively.

It was demonstrated in [4] that all the second encoder blocks of SA-EEND [2], RX-EEND [3] and their proposed model show only identity-like attention weight matrices on two SA heads, for example, in head2 and head3. Interestingly, the two attention weight matrices (head2 and head3) of the second block of SA-EEND, subjected to the proposed $\mathcal{L}_{H2H}$ loss, exhibited non-identity-like patterns in addition to the identity-like patterns in Figure 2(e). When applying $\mathcal{L}_{H2H}$ only to the first block, the training of SA heads in the middle blocks may be impacted, because it minimizes the MSE between the SA heads in the first block and the SA heads in the upper blocks. Therefore, the identity matrix has been relaxed, and performance improvement could have been achieved by effectively utilizing the redundantly trained SA heads.

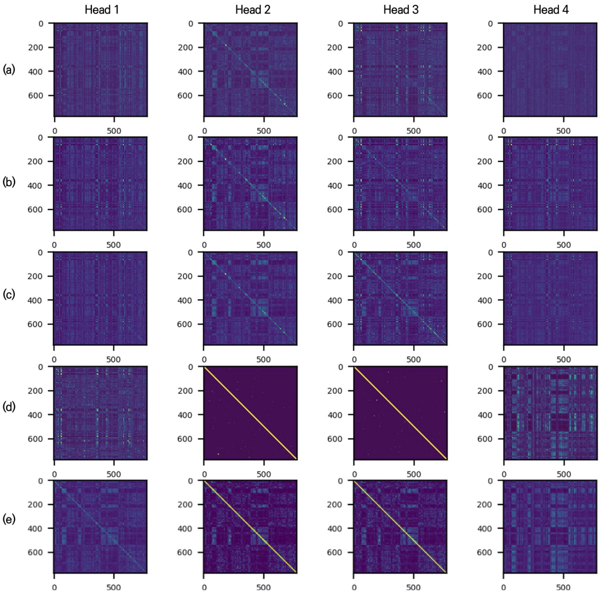

Figure 2: *Visualization of attention weight matrices of (a)–(c) 1st encoder block and (d)–(e) 2nd encoder block: (a) SA-EEND, (b) SA-EEND fine-tuned with $\mathcal{L}_{O2H}$ (c) SA-EEND fine-tuned with $\mathcal{L}_{H2H}$ (d) SA-EEND, (e) SA-EEND fine-tuned with $\mathcal{L}_{H2H}$ (CH recording ID: iaai)*

6. Conclusions

In this study, we proposed SD techniques for the SA-EEND model to increase the contribution of SA heads of lower transformer encoder blocks by sharing high-level information. We considered that the lower the similarity between the matrix to distill and the attention weight matrix to be distilled, the lower the contribution of the attention weight matrix to the diarization performance. To verify this, we assigned SD losses to SA heads that showed low similarity. In particular, our experimental results showed that introducing various SD methods in the fine-tuning step can significantly improve the performance of the SA-EEND model. Furthermore, we visualized the SA heads to which the SD was applied and demonstrated the effects. In future work, we will apply the proposed SD methods for more speakers and deeper networks, although the study in this paper is limited to the four-layered encoder-based structures and two-speaker conversations.

7. Acknowledgment

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) under the artificial intelligence semiconductor support program to nurture the best talents (IITP-2023-RS-2023-00253914) grant funded by the Korea government(MSIT)

8. References

[1] 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*, 2019, pp. 4300–4304.

[2] Y. Fujita, N. Kanda, S. Horiguchi, Y. Xue, K. Nagamatsu, and S. Watanabe, "End-to-end neural speaker diarization with self-attention," in *Proc. IEEE Automatic Speech Recognition and Understanding Workshop*, 2019, pp. 296–303.

[3] Y. Yu, D. Park, and H. K. Kim, "Auxiliary loss of transformer with residual connection for end-to-end speaker diarization," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022, pp. 8377–8381.

[4] Y. R. Jeoung, J. Y. Yang, J. H. Choi, and J. H. Chang, "Improving transformer-based end-to-end speaker diarization by assigning auxiliary losses to attention heads," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2023, pp. 1–5.

[5] L. Zhang, J. Song, A. Gao, J. Chen, C. Bao, and K. Ma, "Be your own teacher: Improve the performance of convolutional neural networks via self distillation," in *Proc. IEEE/CVF International Conference on Computer Vision*, 2019, pp. 3713–3722.

[6] Q. Xu *et al.*, "Self-distillation based on high-level information supervision for compressing end-to-end asr model," in *Proc. INTERSPEECH*, 2022, pp. 1716–1720.

[7] J. Xue, C. Fan, J. Yi, C. Wang, Z. Wen, D. Zhang, and Z. Lv, "Learning from yourself: A self-distillation method for fake speech detection," *arXiv:2303.01211*, 2023.

[8] M. Tzelepi, N. Passalis, and A. Tefas, "Probabilistic online self-distillation," *Neurocomputing*, vol. 493, pp. 592–604, 2022.

[9] Z. Ren, T. T. Nguyen, Y. Chang, and B. Schuller, "Fast yet effective speech emotion recognition with self-distillation," *arXiv:2210.14636*, 2022.

[10] J. S. Seong, J. H. Choi, J. H. Kyung, Y. R. Jeoung, and J. H. Chang, "Noise-aware target extension with self-distillation for robust speech recognition," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2023, pp. 1–5.

[11] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv:1607.06450*, 2016.

[12] A. F. Agarap, "Deep learning using rectified linear units (ReLU)," *arXiv:1803.08375*, 2018.

[13] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Advances in Neural Information Processing Systems*, vol. 30, 2017, p. 5998–6008.

[14] O. Hinton, G.and Vinyals and J. Dean, "Distilling the knowledge in a neural network," in *Proc. NIPS Deep Learning and Representation Learning Workshop*, 2015.

[15] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "SWITCHBOARD: Telephone speech corpus for research and development," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 1992, pp. 517–520.

[16] M. P. Alvin and A. Martin, "NIST speaker recognition evaluation chronicles," in *Proc. Odyssey*, 2004.

[17] A. Martin, M. Przybocki, and J. P. Campbell, "The NIST speaker recognition evaluation program," in *Biometric Systems*, 2005, pp. 241–262.

[18] M. A. Przybocki, A. F. Martin, and A. N. Le, "NIST speaker recognition evaluation chronicles-part 2," in *Proc. Odyssey*, 2006, pp. 1–6.

[19] A. F. Martin and C. S. Greenberg, "NIST 2008 speaker recognition evaluation: Performance across telephone and room microphone channels," in *Proc. INTERSPEECH*, 2009, pp. 2579–2582.

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

[21] A. F. Martin and M. A. Przybocki, "2000 NIST speaker recognition evaluation," in *Philadelphia: Linguistic Data Consortium*, 2001.

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

[23] J. G. Fiscus, J. Ajot, and J. S. Garofolo, "The rich transcription 2007 meeting recognition evaluation," in *Multimodal Technologies for Perception of Humans*, 2007, pp. 373–389.