Long Audio File Speaker Diarization with Feasible End-to-End Models

Kai-Wei Huang* and Chia-Ping Chen†
* National Sun Yat-Sen University, Taiwan
E-mail: kaiwei890604@gmail.com
† National Sun Yat-Sen University, Taiwan
E-mail: cpchen@cse.nsysu.edu.tw

***Abstract*—End-to-End Neural diarization (EEND) systems have become increasingly popular in recent years. However, EEND systems that use permutation invariant training (PIT) losses lead to unpredictable speaker order, making the processing of long audio files challenging. In this paper, we replace the common Transformer encoder in EEND with a Longformer architecture to enhance the model's computational efficiency. We further refine the EEND model with a retention network (RetNet) architecture, achieving RNN-like characteristics without compromising accuracy. Evaluated on the Librimix dataset and our own recording, the Longformer encoder can feasibly diarize audio files 5 times longer than the baseline Transformer encoder. Moreover, the RetNet-EEND module even achieves 100 times longer than the baseline module. In practice, our implementation of RetNet-EEND-EDA model for speaker diarization can easily inference 5 hours long audio file without compromising recognition performance.**

I. Introduction

Speaker diarization is a task to separate different speakers in an audio recording. It is used in many scenarios such as meeting recordings and automatic subtitling of videos. Diarization is commonly used with Automatic Speech Recognition (ASR) to solve the problem of who spoke when and what was said. The traditional method treats diarization as a clustering problem [1], [2]. Voice Activity Detector (VAD) will be used to detect the active signal in the audio frames, which will be sent into a model to extract speaker embedding such as i-vector [1], [2], x-vector [3], [4], and d-vector [5], [6]. Speaker activities will be obtained by utilizing a clustering algorithm for speaker embedding. However, traditional diarization has several problems, such as its inability to handle overlap situations, which means that only one speaker can be active at the same time. Additionally, the clustering algorithm is not trainable so it can not be optimized to the best performance. Therefore, End-to-End Neural Diarization (EEND) [7], a neural network model based on an end-to-end architecture, is proposed as a solution.

EEND treats diarization as a multi-label classification problem by predicting the activity probability of different speakers at different times. Therefore, EEND can easily detect overlap situations, which is not possible with clustering methods. However, end-to-end models also have several limitations, such as handling an unknown number of speakers and the label permutation problem. These issues are not encountered in traditional clustering-based diarization methods. To address the unknown number of speakers, the EEND-EDA architecture introduces the Encoder Decoder Attractor (EDA) [8] framework. For the label permutation issue, EEND employs a permutation-free loss [7], [9] to dynamically determine the order of labels during training. The inconsistencies of the speaker's order and the self-attention mechanism inherent in the Transformer encoder [10], introduce significant challenges in processing long audio files. Consequently, in this paper, we replace the Transformer encoder in EEND to Longformer [11] architecture to reduce the computation. Additionally, we propose an improved RetNet-EEND architecture based on the retention network (RetNet) [12] to address the challenges of recognizing extremely long audio files.

The Longformer architecture enhances the ability to process longer audio files by reducing the complexity of self-attention calculations [10]. It significantly reduces the original computational complexity of self-attention from $O(n^2)$ to $O(n)$. Compared to the traditional Transformer, Longformer significantly reduces computational complexity. However, it still has limitations in processing audio lengths, which cannot meet our requirements for recognizing meeting audio files. Therefore, we propose the RetNet-EEND architecture, which is based on the retention network. The retention mechanism modifies the Transformer architecture by drawing inspiration from the characteristics of RNN models [13]. Traditionally, Transformers consider the tokens of the entire sequence length when computing attention. In contrast, the retention mechanism maintains only a single state for attention calculation. This approach allows for recurrent processing during inference. Through mathematical simplification, it still supports parallel computation during training. RetNet achieves a well-balanced trade-off in terms of computational speed, performance, and memory usage. Therefore, in this paper, we modify the Transformer encoder in the EEND-EDA model to incorporate RetNet. This modification enables the application of RetNet in diarization tasks, demonstrating its effectiveness and efficiency in handling such complex processes. In the RetNet encoder, to address the requirements of processing long conference audio files, we implement a Chunkwise Retention approach. This method involves segmenting the audio into chunks. Within each chunk, parallel computation is employed to calculate attention. Between chunks, a recurrent approach is used, retaining the state of each chunk. Finally, the final retention is

composed by cross chunk retention and inner chunk retention.

In this paper, the second section introduces the baseline architecture of EEND-EDA, along with the Longformer architecture that we aim to compare. We will also provide a detailed introduction to the RetNet-EEND architecture. In the third section, we will describe the datasets used in our experiments, parameter settings, and the performance metrics employed for comparison. The fourth section will present the experimental results in a tabular format and discuss our observations.

II. Model

In this section, we will introduce the baseline End-to-End Neural Diarization with Encoder-Decoder Attractor (EEND-EDA) architecture, Longformer-based encoder EEND, and the chunkwise forward RetNet-EEND architecture model.

A. Baseline Model

Our model is based on the End-to-End Diarization with Encoder-Decoder Attractor (EEND-EDA) architecture. EEND-EDA approaches diarization as a multi-label classification problem. The primary objective of EEND is to predict the speech activity of different speakers at various time points. When the probability at a given time point exceeds a certain threshold, EEND determines that someone is speaking at that moment. Consequently, the output of EEND is a probability distribution with dimension $S \times T$, where S represents the total number of speakers, and T is the temporal dimension of the predicted audio file.

A significant limitation of the traditional EEND model is the need to predetermine the number of speakers during the training phase. In order to address the issue of an unknown number of speakers, the Encoder Decoder Attractor (EDA) architecture was developed. The EDA module primarily utilizes a LSTM encoder-decoder architecture to predict the speaker's number S. The LSTM decoder, using the last cell state and hidden state of the encoder along with a zero vector, generates a variable number of attractors to form an attractor pool. These attractors are processed through linear and sigmoid functions to obtain their respective probabilities of existence. The EDA module will only utilize the top S attractors with none zero probabilities. Those attractors $A \in \mathbb{R}^{D_{\text{model}} \times S}$ are transposed and multiplied with the output of EEND Tranformer's encoder $e_t \in \mathbb{R}^{D_{\text{model}} \times T}$ to obtain the final speaker activity probability $p_t \in \mathbb{R}^{S \times T}$, where D_{model} is the hidden dimension in the model. The final calculation formula for p_t is as follows

$$p_t = \sigma\left(A^{\mathsf{T}} e_t\right) \in (0,1)^S \tag{1}$$

This approach effectively resolves the issue of an unknown number of speakers, which is a significant advancement over the traditional EEND model.

B. Longformer

In practical applications, due to the reliance of EEND on the Transformer architecture, the GPU's VRAM usage increases quadratically with the length of the audio file. Additionally, the permutation-free loss used in EEND results in an unpredictable order of speakers in the output, making it impractical to process the audio by segmentation. Long audio files segmented into different sections may cause inconsistencies in speaker output, leading to processing difficulties. Therefore, we attempted to incorporate the Longformer architecture into the EEND framework. Longformer simplifies the full attention mechanism in self-attention by employing three types of attention mechanisms: sliding window attention, dilated sliding window attention, and global + sliding window attention. In Figure 1, we show the difference between these three methods and full attention.

- Sliding Window Attention:
  This mechanism calculates attention only for tokens near the current token, significantly reducing the computational load compared to full attention, which considers all tokens.
- Dilated Sliding Window Attention:
  Similar to sliding window attention, this approach skips certain tokens to calculate attention, allowing for a broader scope of attention without significantly increasing computational complexity.
- Global + Sliding Window Attention:
  In this mechanism, Longformer selects specific tokens based on the task at hand to receive global attention. This allows certain key tokens to have a broader context, enhancing the model's overall understanding and performance.

By integrating these three types of attention, Longformer significantly reduces the computational complexity of self-attention without a substantial loss in performance. This makes it an efficient solution for processing longer audio files, addressing the limitations of the traditional Transformer-based EEND model in terms of VRAM usage. In this paper, the Longformer architecture is employed, utilizing the Sliding Window Attention and Dilated Sliding Attention mechanisms to construct local and global Attention, respectively. This replaces the Transformer architecture originally used in the EEND-EDA framework, reducing the system's overall complexity. We call this system as Longformer-EEND-EDA.

C. Retention Network

Although the computation is significantly reduced after incorporating Longformer, there still remains an upper limit to the length of audio files it could process. In this paper, we have adopted the retention network (RetNet) to replace the original Transformer architecture, which is inspired by its implementation in large language models (LLMs).

RetNet primarily draws from the architecture of RNNs. Given an input $X \in \mathbb{R}^{|x| \times D_{\text{model}}}$, where D_{model} is the hidden dimension in the model. During inference, it maintains a single state S_n and uses this state to calculate the output with attention. The output O_n is defined as follows

$$\begin{aligned} O_n &= Q_n S_n, & Q_n &\in \mathbb{R}^{1 \times D} \\ S_n &= A S_{n-1} + K_n^{\mathsf{T}} V_n, & A &\in \mathbb{R}^{D \times D}, K_n \in \mathbb{R}^{1 \times D} \end{aligned} \tag{2}$$

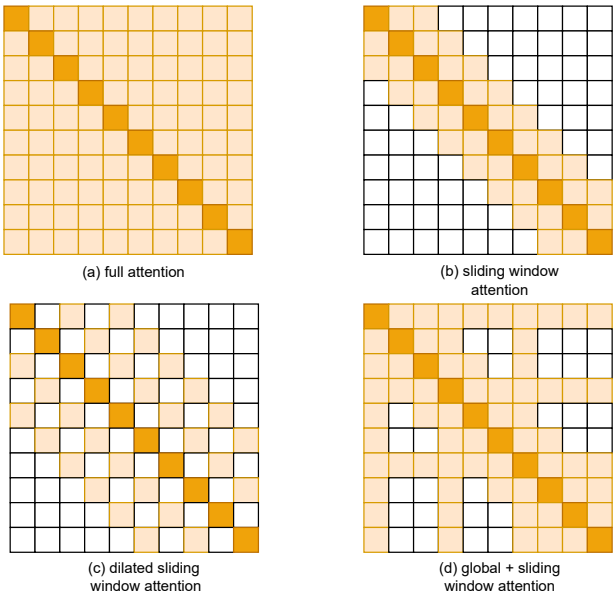

Fig. 1. Comparison between the full self-attention pattern and the attention patterns configured in Longformer.

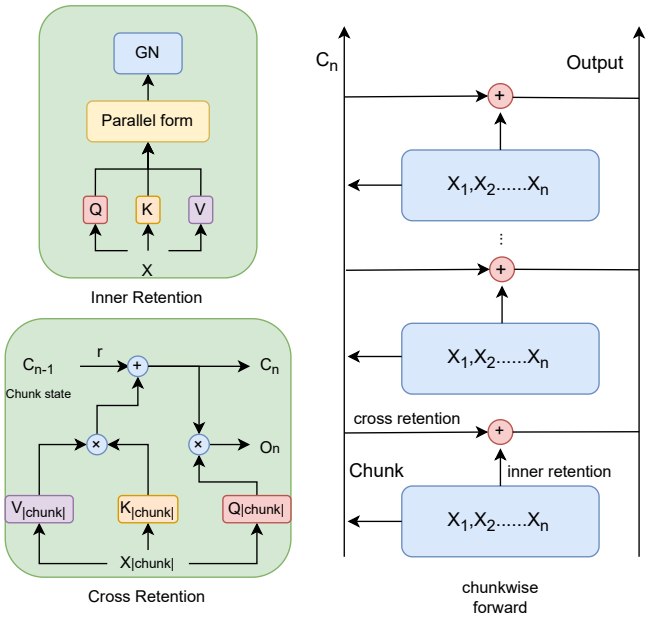

Fig. 2. RetNet chunkwise forward architecture within EEND. We segment the mel spectrogram into 5-second chunks before processing them through the RetNet encoder. Within each chunk, computations are performed in parallel form, while across chunks, we maintain a chunk state C_n to facilitate attention calculations between different chunks.

Q_n,V_n,K_n are simplified projection of X. By expanding the expression of O_n, we can obtain the recurrent representation as follows when S_0 is defined as zero

$$\sum_{m=1}^{n} Q_n A^{n-m} K_m^{\mathsf{T}} v_m \tag{3}$$

Next, diagonalize the matrix A to $p^{-1}\gamma e^{i\theta}p$ then absorbing p^{-1} and p into Q and K, and acknowledging the complex components $\gamma e^{i\theta}$ as Xpos [14] Θ, we can further simplify the recurrent form as follows

$$\text{Parallel Retention}(X) = (QK^{\mathsf{T}} \odot D)V \tag{4}$$

where D is a time exponential decay, and Q K are simply projection from X and dot product Θ. This approach eliminates the need to calculate the complexity of the entire sequence of tokens. During the training phase, the recurrent form can be simplified into a parallel form, significantly enhancing the training performance. This method combines the convenience of parallel training with the low-cost benefits of recurrent inference, effectively capturing the advantages of both approaches.

In the task of diarization, we employ a chunkwise forward representation in Figure 2 to handle extremely long audio files. And the full RetNet-EEND-EDA architecture with chunkwise forward is shown in Figure 3. The audio signal is segmented into various sizes of chunks. Within each chunk, the attention mechanism is calculated in a parallel manner, while between chunks, a recurrent approach is used, retaining only the state of previously computed chunks. The attention between different chunks is calculated using the chunk state, and this inter-chunk attention is then combined with the inner-chunk attention to form the final retention.

III. Experimental Setup

In this section, we will introduce the dataset and the configuration settings used in this experiment. Finally, we will introduce the performance metrics, which is used in our experiments.

A. Dataset

In our experiments, we use Librimix dataset [15], which is mixed by the Librispeech dataset and noise from WHAM! [16]. In this paper, we use Libri2mix only in our experiments, which contain just two speakers. Libri2mix consists of 2,484 speakers, including 1,450 males and 1,034 females. The Librimix dataset is mainly composed of 8khz conversational audio, 58 hours of data for training, and 11 hours of data for validation and testing. We also fine-tuned the EEND-EDA model with the Libri3Mix training set for the multi-speaker task. Our training set consists of a total of 40 hours, while the test and validation sets comprise 11 hours.

In addition to mixed datasets like LibriMix, we also used the ALi-meeting [17] dataset to fine-tune and test the performance on real audio files, and finally evaluated the Retnet-EEND-EDA's performance on long audio recordings. ALi-meeting is a Chinese diarization training set containing a total of 118 hours of annotated audio files, recorded using an 8-channel microphone array. However, in our training, we standardized the audio files to 8kHz and converted them to mono audio to reduce the computational load for end-to-end model training. When training the EEND-EDA-based model, the dataset must

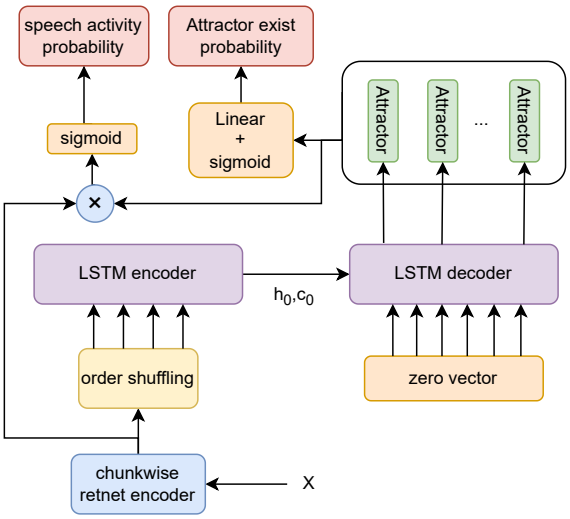

Fig. 3. The overall architecture of RetNet-EEND. The output from the RetNet encoder is sent into the EDA (Encoder Decoder Attractor) module to calculate the number of speakers. Finally, the attractors are multiplied with the original embeddings to obtain the probabilities of speaker activity.

have a uniform number of speakers to proceed. Therefore, we segmented the ALimeeting dataset, detecting 5-second segments within the training set and selecting 10 to 15-second clips containing two speakers for training. After processing, our training set totaled approximately 7 hours, with one hour each for the test and validation sets. Finally, in the data preparation phase, we use the Kaldi recognition toolkit [18] for preprocessing.

B. Configurations

This paper is primarily based on the EEND-EDA architecture from the ESPnet toolkit [19] on Pytorch framework [20]. In the baseline model, all parameter settings remain as original configuration. The Transformer encoder consists of four layers, each layer initially converting an 80-dimensional input into a 512-dimensional representation. In the multi-head attention mechanism, the number of heads is set to four, the final output size is configured to 256, and the dropout rate [21] is set at 0.1. In the EDA module, we maintain the use of an RNN architecture for the encoder-decoder, stacking only a single layer with 256 units. Regarding the Longformer, its overall structure is similar to the Transformer, consisting of four layers in total. For the RetNet, the encoder is also stacked in four layers. In terms of chunk segmentation, we divide 5-second audio clips into individual chunks for the model to process. All three architectures were trained on the Libri2Mix dataset, undergoing a total of 250 epochs with an early stop mechanism.

C. Performance Metrics

In diarization, we use the Diarization Error Rate (DER) [22] as a measure to evaluate performance. Since the results generated by EEND-EDA are discrete, we apply a median filter to smooth the output. In this experiment, we primarily employ an 11-frame median filter with a threshold of 0.5 and a collar tolerance of 0.0 seconds for evaluation purposes.

IV. Results

In this section, we will explore the Longformer-EEND-EDA and the RetNet-EEND-EDA architectures proposed in this paper. We will test these two models, which are improved for handling long audio files, on the test set of the Libri2Mix to evaluate their Diarization Error Rate (DER). These results will be compared with Transformer which is used full self-attention. Next, we will use our own recorded audio files to test the limits of recognition length for different architectures and present the final results in Table I and Table II.

In the baseline EEND-EDA model tested on the two-speaker Libri2Mix dataset, the DER was approximately 7.03. In the Libri3Mix fine-tuning tests, the EEND-EDA model achieved a DER of approximately 11.95. With a complete self-attention mechanism, the performance reveals commendably low error rates. However, this comes at the cost of significantly increased computational complexity. In our tests based on Nvidia GTX-1080Ti for inference, the EEND-EDA model was only capable of processing audio files up to three minutes in length. This duration is insufficient for typical audio recordings.

In contrast, the Longformer-EEND-EDA model, by simplifying the self-attention complexity from $O(n^2)$ to $O(n)$, effectively reduces the overall computational complexity. However, this simplification results in a slight decrease in performance, with the DER increasing from 7.03 for the Transformer to 10.32, marking an increase of 3.29. But the recognizable length of audio files saw a substantial improvement, extending from three minutes to forty-five minutes. In contrast, the Longformer model achieved a DER of approximately 11.75 in the Libri3Mix fine-tuning tests. For the three-speaker fine-tuning, the Longformer performed better compared to EEND-EDA, which we attribute to its relatively simplified model architecture. Without affecting the inference speed, the Longformer greatly extends the length of the audio files that can be recognized.

In the final phase of our research, we integrated the RetNet into the EEND model architecture, leveraging the characteristics of RNNs within RetNet to transition EEND towards a recurrent recognition approach. This modification enables EEND to process audio files of extremely long length. With the simplification of the recurrent form in RetNet, the model retains the advantages of the attention mechanism. Our experiments explored two distinct Xpos strategies: a global Xpos and a chunkwise Xpos calculation. With the global Xpos configuration, the model could understand global positional

relationships. However, this approach became impractical for excessively long audio files. Consequently, we refined the Xpos calculation method to only consider the positional encoding within each chunk in a chunkwise forward manner. This adjustment had a minimal impact on performance but significantly increased the length of audio files that could be processed. The global Xpos version encountered difficulties with audio files exceeding 15 minutes, whereas the revised chunkwise Xpos approach allowed us to process audio files up to three hours in length, with hardware utilization approximately at 60%. Since the model incorporating Global Xpos did not perform optimally on longer audio files, we chose not to proceed with testing on other datasets. We estimated that on GTX-1080Ti, it could handle audio files close to five hours in length, which we believe meets the vast majority of user needs. In GTX-1080Ti, EEND-EDA and Longformer architecture cost about 12 hours for training. The training of RetNet-EEND-EDA took nearly an entire day due to its recurrent mechanism.

In terms of recognition accuracy, the unique design of RetNet even surpassed the performance of traditional Transformer models. The DER achieved with global Xpos was 5.8, and with the transition to chunkwise Xpos, the DER slightly increased to 6.16. RetNet with chunkwise Xpos architecture successfully balances practicality and accuracy, indicating its substantial potential and utility for processing long audio files in diarization tasks. The Retnet-EEND model achieved a DER of approximately 10.75 in the Libri3Mix fine-tuning tests. Compared to the Transformer and Longformer models, the Retnet architecture demonstrated superior learning ability, reducing the error rate by about 1 relative to the previous systems.

In the ALi-meeting real recording test phase, we used a custom pre-processing program to extract segments with fixed two-speaker scenarios from the ALi-meeting dataset for fine-tuning. This significantly reduced the size of the overall dataset. However, in our tests, the Retnet-EEND-EDA model achieved a DER of 22.61 on our custom test set. This performance is notably better than the Transformer model's fine-tuning result of 25.75 on the ALi-meeting dataset. Furthermore, we tested the Retnet-EEND-EDA model on several long audio files from the original ALi-meeting test set, and the results were consistent with the fine-tuning performance. This demonstrates that the Retnet-EEND-EDA model surpasses the baseline Transformer model both in terms of recognition accuracy and the ability to handle longer audio files.

Although RetNet significantly improves accuracy and recognition length compared to Longformer and Transformer, its recognition speed is a drawback. Due to RetNet's recurrent design, it inherently cannot compete with the parallel computation capabilities of Longformer and Transformer in terms of speed. In our tests, recognizing a one-minute audio file with each chunk segmented into five seconds took approximately one second, whereas the Transformer and Longformer architectures required only about one-tenth of that time. We believe the RetNet architecture may be more suitable for non-real-time recognition scenarios.

TABLE I
PERFORMANCE OF OUR PROPOSED MODELS WITH DIFFERENT ARCHITECTURES. INCLUDING INFERENCE MAXIMUM LENGTH, AND TIME EACH MODEL COSTS WHEN INFERENCE 1 MINUTE AUDIO IN GTX 1080TI WITH 11GB VRAM.

Architecture	Length(↑)	Speed(↓)
Transformer	3 min	**0.11s/min**
Longformer	45 min	0.14s/min
RetNet global Xpos	15 min	1s/min
RetNet chunkwise Xpos	**300 min**	1s/min

TABLE II
DIARIZATION ERROR RATE (DER) PERFORMANCE OF OUR PROPOSED MODELS WITH DIFFERENT ARCHITECTURES IN DIFFERENT DATASETS. INCLUDING LIBRI2MIX, LIBRI3MIX, AND FINE-TUNING ON ALI-MEETING DATASET WITH PREPROCESSING

Architecture	Libr2mix(↓)	Libri3mix(↓)	ALi-meeting(↓)
Transformer	7.03	11.95	25.75
Longformer	10.32	11.75	27.34
RetNet global Xpos	**5.8**	-	-
RetNet chunkwise Xpos	6.16	**10.7**	**22.61**

V. CONCLUSIONS

In this paper, we propose two architectures, Longformer-EEND-EDA and RetNet-EEND-EDA, to mitigate the limitations of end-to-end diarization in recognizing long audio files. In the Longformer-EEND model, we leverage the Longformer's characteristic to reduce the computational load of self-attention within the Transformer framework, thereby decreasing the complexity of the EEND architecture. This adaptation enables the Longformer-EEND-EDA to extend the recognition length by 15 times compared to the baseline EEND-EDA model. However, due to the reduction in attention computation, the overall recognition accuracy decreased, and the recognition length still did not meet our requirements. Consequently, we explored incorporating the concept of the retention network into EEND, achieving promising results. After incorporating the retention network, due to RetNet's emulation of RNN characteristics and its unique attention mechanism calculations, the overall recognition performance improved by 100 times compared to the baseline EEND-EDA model. Moreover, the accuracy of RetNet-EEND-EDA slightly improved compared to the full attention EEND-EDA model. With these enhancements, RetNet-EEND-EDA significantly improves recognition length while maintaining a practical inference speed, alongside an increase in accuracy.

REFERENCES

[1] G. Sell and D. Garcia-Romero, "Speaker diarization with plda i-vector scoring and unsupervised calibration," in *2014 IEEE Spoken Language Technology Workshop (SLT)*, IEEE, 2014, pp. 413–417.

[2] S. H. Shum, N. Dehak, R. Dehak, and J. R. Glass, "Unsupervised methods for speaker diarization: An integrated and iterative approach," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 10, pp. 2015–2028, 2013.

[3] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker diarization using deep neural network embeddings," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2017, pp. 4930–4934.

[4] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2018, pp. 5329–5333.

[5] Q. Wang, C. Downey, L. Wan, P. A. Mansfield, and I. L. Moreno, "Speaker diarization with lstm," in *2018 IEEE International conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2018, pp. 5239–5243.

[6] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2018, pp. 4879–4883.

[7] Y. Fujita, N. Kanda, S. Horiguchi, K. Nagamatsu, and S. Watanabe, "End-to-end neural speaker diarization with permutation-free objectives," *arXiv preprint arXiv:1909.05952*, 2019.

[8] 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," *arXiv preprint arXiv:2005.09921*, 2020.

[9] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2017, pp. 241–245.

[10] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[11] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The long-document transformer," *arXiv preprint arXiv:2004.05150*, 2020.

[12] Y. Sun, L. Dong, S. Huang, *et al.*, "Retentive network: A successor to transformer for large language models," *arXiv preprint arXiv:2307.08621*, 2023.

[13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[14] Y. Sun, L. Dong, B. Patra, *et al.*, "A length-extrapolatable transformer," *arXiv preprint arXiv:2212.10554*, 2022.

[15] J. Cosentino, M. Pariente, S. Cornell, A. Deleforge, and E. Vincent, "Librimix: An open-source dataset for generalizable speech separation," *arXiv preprint arXiv:2005.11262*, 2020.

[16] G. Wichern, J. Antognini, M. Flynn, *et al.*, "Wham!: Extending speech separation to noisy environments," *arXiv preprint arXiv:1907.01160*, 2019.

[17] F. Yu, S. Zhang, Y. Fu, *et al.*, "M2met: The icassp 2022 multi-channel multi-party meeting transcription challenge," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2022, pp. 6167–6171.

[18] D. Povey, A. Ghoshal, G. Boulianne, *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, IEEE Signal Processing Society, 2011.

[19] S. Watanabe, T. Hori, S. Karita, *et al.*, "Espnet: End-to-end speech processing toolkit," *arXiv preprint arXiv:1804.00015*, 2018.

[20] A. Paszke, S. Gross, F. Massa, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[21] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[22] J. G. Fiscus, J. Ajot, M. Michel, and J. S. Garofolo, "The rich transcription 2006 spring meeting recognition evaluation," in *Machine Learning for Multimodal Interaction: Third International Workshop, MLMI 2006, Bethesda, MD, USA, May 1-4, 2006, Revised Selected Papers 3*, Springer, 2006, pp. 309–322.