Interspeech 2025
17-21 August 2025, Rotterdam, The Netherlands

End-to-End Diarization utilizing Attractor Deep Clustering

David Palzer[1], *Matthew Maciejewski*[2], *Eric Fosler-Lussier*[1]

[1]Computer Science and Engineering, The Ohio State University, U.S.A.
[2]Human Language Technology Center of Excellence, The Johns Hopkins University, U.S.A.
palzer.1@osu.edu, mmaciej2@jh.edu, fosler@cse.ohio-state.edu

Abstract

Speaker diarization remains challenging due to the need for structured speaker representations, efficient modeling, and robustness to varying conditions. We propose a performant, compact diarization framework that integrates conformer decoders, transformer-updated attractors, and a deep clustering style angle loss. Our approach refines speaker representations with an enhanced conformer structure, incorporating cross-attention to attractors and an additional convolution module. To enforce structured embeddings, we extend deep clustering by constructing label–attractor vectors, aligning their directional structure with audio embeddings. We also impose orthogonality constraints on active attractors for better speaker separation while suppressing non-active attractors to prevent false activations. Finally, a permutation invariant training binary cross-entropy loss refines speaker detection. Experiments show that our method achieves low diarization error while maintaining parameter count.

Index Terms: speaker diarization, conformer, attractors, DPCL, angle loss, orthogonality constraints

1. Introduction

Speaker diarization, the task of segmenting an audio stream into homogeneous speaker segments, remains a challenging problem due to the inherent variability in acoustic conditions, the presence of overlapping speech, and the need to robustly handle a wide range of speakers [1, 2]. End-to-end neural diarization systems (EEND) [3] allow for direct minimization of diarization errors during training. The original EEND included an intermediate auxiliary loss of Deep Clustering (DPCL) [4] to help discriminate speakers; DPCL itself was introduced as a method to explicitly create discriminable embedding vectors for an arbitrary number of classes and was evaluated on speech separation.

Further work in the EEND line led to Encoder-Decoder Attractors (EDA) [5] where an EDA module was responsible for encoding discriminable speaker representations directly from audio embeddings, allowing for a flexible number of speakers to be separated. Neither this work nor subsequent EEND studies (e.g., [6, 7]) utilized DPCL.

In this work, we propose a performant, compact diarization framework that synthesizes the strengths of conformer-based decoders, transformer-updated attractors, and an improved DPCL-style angle loss to promote robust, structured representations. By enhancing a conformer architecture with cross-attention to attractors and incorporating additional convolutional modules, we capture both local and global speaker information while keeping the network size manageable. We also reintroduce DPCL loss into the EEND framework (first seen in [3] but absent in subsequent EEND models) with a new extension to enforce directional alignment between label–attractor vectors and audio embeddings, leveraging orthogonality constraints on active attractors to reduce cross-speaker confusion. Finally, we include a permutation invariant training (PIT) [8] binary cross-entropy (BCE) loss term to refine speaker detection, effectively suppressing non-active attractors and minimizing false alarms.

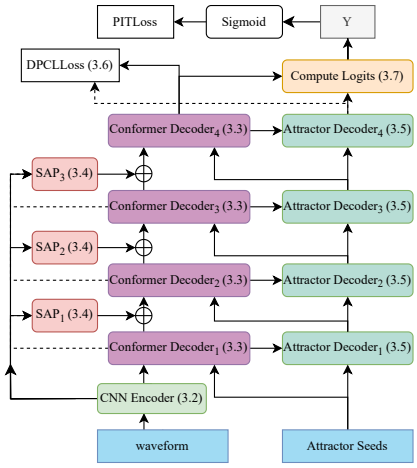


Figure 1: *Example four layer EEND-CD architecture. Each block is described in subsections of Section 3.*

Experiments demonstrate that our approach achieves competitive or superior diarization error rates (DER), while maintaining efficiency due to the compactness of the architecture. The key contributions of this paper are:

- An **enhanced conformer decoder** design that integrates Latte based attention [9] and an attractor-based cross-attention for refined speaker representation.
- An improved **DPCL-style angle loss extension** using label–attractor vectors to enforce structured embedding alignments and orthogonality constraints.
- A novel **speaker detection refinement** strategy that explicitly works to reduce false attractor activations.

The rest of the paper is organized as follows. In Section 2, we discuss related studies in speaker diarization and representation learning. Our proposed method, including the conformer architecture, attractor updating mechanism, and extended DPCL loss, is described in Section 3. Section 4 details experimental setups and datasets, and Section 5 presents our results, discussion, and future directions.

10.21437/Interspeech.2025-2030

2. Related Work

Speaker diarization involves partitioning an audio stream according to speaker identity, a task that has traditionally relied on statistical modeling and clustering approaches. Early methods often combined Gaussian Mixture Models (GMM) or i-vector [10] representations with clustering [1, 11], later transitioning to x-vectors [12] for more robust speaker embeddings. According to Park *et al.*, [2], "many traditional speaker diarization systems, especially clustering-based systems, have only focused on non-overlapping regions and even the overlapping regions are excluded in the evaluation metric. (pg 22)"

Deep learning-based strategies have accelerated progress in speech and diarization [2], especially with the introduction of End-to-End Neural Diarization (EEND) [3]. DPCL projects time-frequency bins (or time frames) into an embedding space, then applies k-means to cluster an arbitrary number of speakers. In contrast, EEND directly predicts speaker activities in a permutation-invariant manner, but requires speaker counts a priori. More recently, attractor-based approaches [5, 7, 6] introduced learnable speaker prototypes (attractors) that iteratively refine speaker estimates, enabling the EEND models to adapt to an unknown or variable number of speakers. Despite these advances, challenges remain in capturing both local and global acoustic context efficiently, as well as maintaining scalability to longer recordings.

Work incorporating label dependency has been explored by [6], where intermediate attractors are utilized. These attractors are computed as a weighted sum of intermediate diarization predictions and subsequently added to each time frame before being processed by subsequent layers. This approach has been shown to reduce the missed speech rate, albeit at the cost of an increased false alarm rate.

The conformer architecture [13] has proven effective in speech processing tasks by combining local convolutions with global self-attention, improving upon pure transformer-based models in Automatic Speech Recognition (ASR). In diarization, conformer-based encoders have shown promise for generating robust speaker embeddings [7, 14, 15]. However, we are not aware of any diarization works that have explored deep conformer-based decoders that also integrate attractor-based cross-attention.

In parallel, various forms of optimized self-attention have been proposed to reduce quadratic complexity or to focus on latent representations of the sequence [16, 17]. Our work adopts a latent self-attention (Latte [9]) mechanism within each conformer decoder, providing an efficient means of capturing global context across frames while maintaining task performance.

Finally, angle-based losses, such as the DPCL-style angle loss, have been shown to encourage discriminable embeddings which show improvements in related fields such as speaker separation [4, 18]. Following papers have pursued this idea further [19] and offered alternative loss formulations.

3. Proposed Method

In this section, we introduce our diarization framework, which combines a CNN-based feature extractor, conformer blocks with latent self-attention (Latte), attractor-based cross-attention, depthwise pooled residual connections, and a DPCL-style angle loss for structured embeddings. Figure 1 illustrates the overall architecture.

3.1. Overview of the Architecture

Our framework takes a batch of recordings as input, where the recordings are processed into 23-dimensional mel features at a 10 ms frame rate. These frames are further stacked into windows of 15 frames using a hop of 10 frames such that each window overlaps with the next by 5 frames. We represent this input as a 4D tensor of shape Batch × Time × Window × Mels. Previous models flatten these into a 3D tensor Batch × Time × (Window × Mels), which is fed to the network directly, however we instead encode these into embeddings using a CNN described in the next section. These encoded embeddings are then passed into a stack of conformer blocks where each block is paired with a transformer decoder that updates speaker attractors. Finally, we compute speaker logits using a binary cross entropy loss in a permutation-invariant manner.

3.2. CNN-based Front End

The CNN treats each Window × Mels slice as a single "grayscale" image with one input channel, $1 \times \text{Window} \times \text{Mels}$. We use a 5-layer CNN, each with a 3×3 kernel, except the last kernel of 1×2. We also increase the number of output filters (16, 32, 64, 128, 256) at each layer and downsample by half. This design preserves local time–frequency relationships and produces a robust representation before feeding into subsequent model components. By the final layer, the temporal dimension, T, remains intact (one embedding per time step), while the spatial dimensions collapse to a single 256-dimensional feature vector, X_t. We then apply an RMSNorm to stabilize the embeddings before feeding them into the conformer blocks:

$$\mathrm{X}_t \in \mathbb{R}^{256}, \quad \forall\, t = 1, 2, \ldots, T. \tag{1}$$

3.3. Latte-based Conformer Decoders

Each conformer block follows the sequence:

1. **Feed-Forward (FF)**: A positionwise two-layer MLP with a residual connection.
2. **Latent Self-Attention (Latte)**: A multi-head latent-attention mechanism applied to the time dimension [9]. This captures global temporal relationships.
3. **CNN**: A depthwise convolution module that captures local context.
4. **Cross-Attention to Attractors**: The output of the CNN is used as queries, while the current speaker attractors serve as keys and values (see Section 3.5).
5. **CNN**: A second convolution block for further refinement.
6. **Feed-Forward (FF)**: Closes the block with another FF sublayer and residual.

Each conformer block outputs Batch × Time × 256, which is then used to compute speaker logits or passed to subsequent layers (Section 3.4). This refinement strategy allows complex interactions between audio embedding and attractors.

3.4. Self-Attentive Pooling (SAP) and Residual Injection

We maintain a list of conformer block outputs across network depth to be used in the self-attentive pooling (SAP) blocks. Suppose we have processed $d-1$ blocks, generating $\{\mathbf{X}^{(0)}, \ldots, \mathbf{X}^{(d-1)}\}$, each of dimension Batch × Time × 256. Before feeding block d, we:

1. Stack these embeddings along a depth axis:

$$\mathbf{H}_{\text{stack}}^{(d-1)} \in \mathbb{R}^{\text{Batch} \times \text{Time} \times (d) \times 256}. \tag{2}$$

2. Pool across this depth dimension using self-attentive pooling, yielding $\mathbf{P}^{(d-1)} \in \mathbb{R}^{\text{Batch}\times\text{Time}\times 256}$.
3. Residually add $\mathbf{P}$ to the current input (the most recent audio embeddings):

$$\mathbf{X}^{(d-1)} = \mathbf{X}^{(d-1)} + \mathbf{P}^{(d-1)}. \tag{3}$$

4. Feed $\mathbf{X}^{(d-1)}$ into the conformer block, d.

This ensures each block sees a global summary of all prior representations, acting as a *global residual* connection, and smooths training convergence.

3.5. Attractor Updates via Transformer Decoder

We initialize a set of attractors $\mathbf{A} \in \mathbb{R}^{S\times E}$ as a set of 256-dimensional vectors. These are *tiled* across the batch as needed. After each conformer block computes its audio representation, a transformer decoder refines the attractors:

- **Self-Attention:** Attractors attend to themselves to capture inter-speaker relationships.
- **Cross-Attention:** The updated attractors then attend to the conformer output for the current layer, enabling them to adapt to the updated audio representations.

Each attractor $\mathbf{a}_s$ thus evolves through the depth of the network, allowing multiple comparisons against it in the conformer.

3.6. Deep Clustering Auxiliary Loss

During initial experimentation we applied the original DPCL [4], however, we found that once a model had learned to detect speech and was learning to differentiate speakers it would diverge. This is most likely due to the mismatch between DPCLs orthogonality constraint and our BCE loss.

Our approach extends the original DPCL formulation [4] in two ways; first, by leveraging multi-class diarization labels we recognize that constraints can be values other than parallel or orthogonal, and second, by allowing attractors themselves to specify the relative directions. These relative angles, in both cases, ensuring consistent geometric structure across training.

Let $\mathbf{X} \in \mathbb{R}^{T\times E}$ be the set of L_2-normalized audio embeddings for T time frames, and $\mathbf{A} \in \mathbb{R}^{S\times E}$ be the set of attractors for S speakers, where E is the embedding dimension. The ground truth label matrix $\mathbf{Y} \in \{1, -1\}^{T\times S}$ indicates speaker activity at each time frame, with 1 for active speakers and -1 for inactive speakers.

First, we propose MultiOpposite DPCL (MO-DPCL). For each time frame t, we construct a label vector:

$$\mathbf{l}_t = \frac{y_t}{\max(||y_t||_2, \epsilon)} \tag{4}$$

The DPCL auxiliary loss is then computed using pairwise embedding distances:

$$\updownarrow_{\text{DPCL}_{i,j}} = (\langle l_i, l_j\rangle - \langle \mathbf{x}_i, \mathbf{x}_j\rangle)^2, \tag{5}$$

where $\langle\cdot,\cdot\rangle$ denotes the inner product. This formulation ensures that embeddings which represent similar sets of speakers are clustered while pairs which differ will point away, saturating in opposite directions rather than orthogonality.

Additionally, we propose Attractor DPCL (A-DPCL). For each time frame t, we construct a label-attractor vector:

$$\mathbf{l}_t = \frac{y_t \cdot \mathbf{A}}{\max(||y_t \cdot \mathbf{A}||_2, \epsilon)} \tag{6}$$

We compute the A-DPCL loss in a pairwise manner as above, and backpropagate though both audio embeddings, and label-attractors. While similar to MO-DPCL, this variant allows for more flexibility in the embedding geometry, and causes the relative geometry of the attractors and audio embeddings to be directly compared and reflected in each other.

These auxiliary losses are applied on the final audio embedding which are used in our logit calculation.

3.7. Logits Computation and Orthogonal Attractor Loss

The last layer's attractors are split into a 256-dimension final attractor, a_s, and a bias term, b_s. We compute speaker logits by taking a dot product between the conformer output $\mathbf{x}_t^{(d)}$ and each attractor vector $\mathbf{a}_s$, adding both the attractor bias b_s and a learned global bias b_{global}:

$$\text{logit}_{t,s} = \mathbf{x}_t^{(d)} \cdot \mathbf{a}_s + b_s + b_{\text{global}}. \tag{7}$$

We apply a sigmoid to obtain per-speaker probabilities, using a suppressive BCE loss that penalizes non-active attractors. Specifically, we push attractors that are not assigned to a speaker to zero exactly, as a clean signal, and only consider the bias terms in the prediction. Using a mean squared error (MSE) loss, we encourage orthogonality among *active* attractors to reduce speaker confusion in overlapping speech.

4. Experiments

4.1. Data Preparation

Following the example of prior EEND studies [6], we prepare data from the CALLHOME and Switchboard-2 (Phase II, III) corpora. Unlike some earlier works, we do not have access to the Switchboard-2 (Phase I) or the NIST Speaker Recognition Evaluation data. Moreover, whereas [6] restricts training to two-speaker mixtures, our system can handle a larger number of speakers. Therefore, we generate training mixtures from the Switchboard-2 (Phase II, III) dataset and the CALLHOME training set. Validation is performed on the CALLHOME validation split, and final evaluations use the CALLHOME test set.

4.2. Acoustic Features

Each audio segment is converted into 23-dimensional log-Mel filterbank features with a frame length of 10 ms. We concatenate 15 consecutive frames (window size) with a 10 ms hop, resulting in 345-dimensional feature vectors that represent 100 ms of audio. These frames serve as inputs to our 5-layer CNN front-end (Section 3.2). During training, we randomly select a 50 s contiguous portion of each recording to accelerate training.

4.3. Augmentation

We employ noise augmentation from the MUSAN dataset [20], injecting background audio to improve robustness. Following [21], we do not use any reverberation augmentation for simplicity. All audio is resampled to 8 kHz prior to feature extraction.

4.4. Training Configuration

Our models are trained for 2000 epochs using AdamW optimization with a one-cycle learning rate scheduler. We use a batch size of 64, which balances efficiency and convergence stability. We select the best-performing checkpoint based on validation loss, then evaluate on the CALLHOME test set.

Model	Parameters		DER (%)				SAD (%)	
	Total	Free	DER	MS	FA	CF	MS	FA
1. EEND-EDA [5]	6.4M	6.4M	9.96	5.40	1.36	2.81	3.85	0.87
2. EEND-EDA-deep [6]	29.3M	17.0M	8.50	4.43	1.31	2.76	3.15	0.85
3. EEND with Attribute Attractors [7] (12 layers)	33.7M	33.7M	7.87	4.18	1.42	2.27	3.28	0.84
4. EEND with Attribute Attractors + Conformer [7] (12 layers)	35.3M	35.3M	6.98	3.63	1.95	1.41	3.68	0.82
5. 7 + SAP + A-DPCL (final system, 5 layers)	15.3M	15.3M	**4.99**	3.60	**1.12**	0.27	**2.45**	**0.47**
Ablation study on changes required to get to final system								
6. EEND with Conformer Decoder (12 layers) + last layer Attractor Decoder + MO-DPCL	22.2M	22.2M	5.78	4.11	1.45	**0.22**	3.05	0.51
7. 6 + per layer Attractor Decoder (reduce to 5 layers)	14.8M	14.8M	5.60	**3.52**	1.72	0.36	2.49	0.68

Table 1: *Diarization error rates (DER) for each model on the CALLHOME test set and component results.*

4.5. Model Configuration

For all experiments, the CNN front-end downsamples the 345-dimensional windowed features to 256-dimensional embeddings, which are RMS-normalized before entering the stack of conformer blocks (Section 3.3). We use 5 layers of conformer blocks and attractor decoder blocks, all of hidden dimension 256, each producing intermediate representations. The latent self-attention (Latte) mechanism operating within each conformer block has a hidden dimension of 128. Speaker attractors, initialized as $\mathbf{8} \times \mathbf{256}$ dimension vectors, are updated by a transformer decoder, enabling our system to track a large number of concurrent speakers.

4.6. Losses and Evaluation

We compute logits for each time frame and speaker attractor, combining a suppressive binary cross-entropy loss with a DPCL-style angle loss (Section 3.7). Evaluation focuses on standard diarization metrics diarization error rate (DER), and speech activity detection (SAD).

5. Results and Discussion

5.1. Evaluation on CALLHOME

Table 1 presents the diarization performance of our proposed system and several baseline models on the CALLHOME test set, measured in terms of diarization error rate (DER), missed speech (MS), false alarm (FA), and confusion (CF). Our final system, which integrates a conformer-based backbone, attractor-based cross-attention, and A-DPCL with self-attentive pooling, achieves a DER of 4.99%, significantly outperforming existing methods.

Compared to the baseline EEND-EDA system [5] (DER 9.96%) and the deeper EEND-EDA-deep model [6] (DER 8.50%), our approach reduces the overall error by nearly 50%. Notably, our system surpasses the recent EEND with Attribute Attractors (7.87%) and its conformer-enhanced variant (6.98%) by a substantial margin, demonstrating that our architectural innovations contribute meaningfully to diarization accuracy.

Looking at DER, our method achieves the lowest confusion rate (0.27%), reflecting the effectiveness of our attractor-based decoding in maintaining clear speaker separation. While maintaining competitiveness in missed speech (3.60%), we see a modest improvement in the false alarm rate (1.12%) over previous systems. The final system also significantly improves in SAD missed speech (2.45%) and false alarms (0.47%), indicating robust detection of both speech and silence regions.

5.2. Original Model and Ablation Study

Our original model, row 6 in Table 1, employed a 12-block conformer decoder integrated with one attractor-based decoder and MO-DPCL loss. This architecture allowed a multi-depth comparison between audio embeddings and static speaker embeddings leading to gains in speaker differentiation, achieving a DER of 5.78% with the lowest confusion rate (0.22%) across all evaluated systems. However, the model's relatively high missed speech (4.11%) and false alarm rate (1.45%) indicated potential inefficiencies in speaker detection.

Reducing the number of conformer blocks from 12 to 5 while incorporating a per-layer attractor decoder improves model size while seeing modest improvements in diarization error (DER 5.60%). Introducing the SAP mechanism, for training efficiency, and A-DPCL loss further enhances speaker separation and reduces DER to 4.99%. We do see a small increase in confusion, however the overall tradeoff is worth it. This analysis underscores the importance of each component, with the the conformer decoder effective at separating speakers, and A-DPCL loss particularly effective in lowering false alarms.

5.3. Analysis and Future Directions

The strong performance of our system can be attributed to several key design choices. The conformer backbone efficiently models local and global dependencies, while the attractor-based decoder ensures consistent speaker embeddings across layers. The addition of SAP facilitates the aggregation of temporal information, and A-DPCL loss enforces alignment between audio embeddings and speaker attractors, resulting in reduced confusion and improved speaker differentiation.

Despite our strong performance, the gains over the previous EEND system in [7] are primarily from decreases in false alarms and confusions; the miss rate remains largely unchanged. We hypothesize that the the use of independent speaker attractors that are encouraged to have orthogonal representations allows the model to better represent regions where there previously were confusions, but a fuller analysis will be left to future work.

Future work will also focus on refining miss rates further and extending our approach to handle more complex multi-speaker environments with minimal computational overhead.

Our findings highlight the importance of balancing model complexity with performance and demonstrate that integrating structured embedding losses and attention mechanisms can significantly advance the state of the art in neural speaker diarization.

6. References

[1] X. Anguera, S. Bozonnet, N. W. D. Evans, C. Fredouille, D. Mason, and E. S. Chng, "Speaker diarization: A review of recent research," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 356–370, 2012.

[2] 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," *Computer Speech & Language*, vol. 72, p. 101317, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0885230821001121

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

[4] J. R. Hershey, Z. Chen, J. L. Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 31–35.

[5] S. Horiguchi, Y. Fujita, N. Kanda, S. Watanabe, and T. Nakatani, "End-to-end speaker diarization for an unknown number of speakers with encoder-decoder based attractors," in *Interspeech 2020*, 2020, pp. 269–273.

[6] Y. Fujita, T. Komatsu, R. Scheibler, Y. Kida, and T. Ogawa, "Neural diarization with non-autoregressive intermediate attractors," in *2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[7] D. Palzer, M. Maciejewski, and E. Fosler-Lussier, "Improving neural diarization through speaker attribute attractors and local dependency modeling," in *2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 11 911–11 915.

[8] 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 Press, 2017, p. 241–245. [Online]. Available: https://doi.org/10.1109/ICASSP.2017.7952154

[9] R. Dolga, L. Maystre, M. Cobzarenco, and D. Barber, "Latte: Latent attention for linear time transformers," 2024. [Online]. Available: https://arxiv.org/abs/2402.17512

[10] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[11] 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.

[12] 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.

[13] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented transformer for speech recognition," in *Interspeech 2020*, 2020, pp. 5036–5040.

[14] T.-Y. Leung and L. Samarakoon, "Robust end-to-end speaker diarization with conformer and additive margin penalty," in *Interspeech 2021*, 2021, pp. 3575–3579.

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

[16] S. Wang, B. Z. Li, M. Khabsa, H. Fang, and H. Ma, "Linformer: Self-attention with linear complexity," 2020. [Online]. Available: https://arxiv.org/abs/2006.04768

[17] DeepSeek-AI and E. A. Liu, "Deepseek-V3 technical report," 2024. [Online]. Available: https://arxiv.org/abs/2412.19437

[18] Y. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-channel multi-speaker separation using deep clustering," in *Interspeech 2016*, 2016, pp. 545–549.

[19] Z.-Q. Wang, J. Le Roux, and J. R. Hershey, "Alternative objective functions for deep clustering," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 686–690.

[20] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," 2015, arXiv:1510.08484v1.

[21] F. Landini, A. Lozano-Diez, M. Diez, and L. Burget, "From simulated mixtures to simulated conversations as training data for end-to-end neural diarization," in *Interspeech 2022*, 2022, pp. 5095–5099.