Speaker-conditioned U-shaped Diarization with Speaker Extraction-guided Enhancement

Ngoc Thuan Tran, Ngoc Chau Hoang, and Quoc Cuong Nguyen

***Abstract*—Speaker diarization demarcates speech segments by speaker, answering the question "who spoke when?". Recently, a promising approach has emerged by integrating speaker diarization with speech separation or speaker extraction, which offers better generalization and requires significantly less pretraining data. Despite this progress, efficiently aligning speaker extraction with the inherent requirements of diarization remains a key challenge. In this paper, we introduce a novel joint network, Speaker-conditioned U-shaped Diarization with Speaker Extraction-guided Enhancement (SUDx), which leverages the U-net architecture and exploits hierarchical speaker representation to enhance the performance of speaker diarization. In addition, SUDx does not rely on explicit speaker identity labels for supervision, allowing it to learn distinctive acoustic characteristics and adapt easily to real-recorded multi-speaker conversations. Furthermore, we introduce a novel inference strategy that effectively handles unknown number of speakers and reduces reliance on large-scale pretraining data. We show that SUDx outperforms competitive baselines for speaker diarization while keep extracting quality speech on the LibriMix dataset. We further assess our proposed approach and our novel strategy on the AMI and AISHELL-4 meeting corpora, experimental results indicate that our model achieved state-of-the-art performance with much less pretraining data.**

***Index Terms*—speaker extraction, speaker diarization, multitalker scenario, LibriMix, AMI**

I. Introduction

SPEAKER diarization plays an important role in a variety of real-world applications. The well-known combination of it and other speech processing systems, such as automatic speech recognition, serves as the foundation for tasks that involve noting or tracking events in multi-talker scenarios. These scenarios span a wide range of domains, including telephone [1], [2], broadcast news [3], meetings [4]–[6] or social events [7]. Specifically, speaker diarization aims to estimate speaker speech activities from an input audio, answering the question "who spoke when?" [8], [9].

A traditional approach for speaker diarization task is clustering-based method [2], [10], [11], which treats the problem as a cascade of modular components. The objective in this paradigm is to optimize each individual component to improve the overall system performance. Although this design is easy to train and naturally handles the unknown number of speakers, it fails to address overlapping speech, which happens commonly in conversational settings, due to the assumption that only one speaker is active at a time. Among modern attempts, two notable approaches have emerged: End-to-End Neural Diarization (EEND) [12]–[14], and Target Speaker Voice Activity Detection (TS-VAD) [15]. EEND reformulates speaker diarization as a frame-wise multi-label classification problem, enabling the model to handle overlapping speech by predicting multiple active speakers at each time frame. TS-VAD adopts a similar frame-wise prediction framework but conditions the output on target speaker embeddings, which serve as references for detecting the activity of specific speakers. Despite achieving remarkable results and establishing a solid position, these methods typically require large amounts of pretraining data and struggle to handle the variable number of speakers as flexibly as traditional approaches [12], [13], [16].

Besides addressing speaker diarization in isolation, several studies have explored the interaction between speaker diarization and speech separation [16]–[18]. This integration can be approached using either multi-stage [18] or joint training models [16], [17]. These works show that diarization and separation are complementary, which leads to better generalization [17], [19]. More recently, USED [19] investigates the integration of speaker diarization and target speaker extraction (TSE). This framework is built upon the architectures of SpEx+ [20] and TS-VAD [15], but accepts raw speech as reference inputs rather than pre-computed speaker embeddings. Specifically, it employs a Speaker Encoder to generate speaker embeddings, which are then integrated with a Separator module to disentangle speakers and simultaneously produce both diarization and separation outputs. As a result, USED has achieved state-of-the-art performance for both speaker diarization and speaker extraction tasks with significantly less pretraining data.

Although the combined approach of speaker diarization with speech separation or speaker extraction has achieved notable results, we argue that an efficient method that aligns with the nature of the diarization task remains underexplored in such integration settings. USED adapted the Separator module from SpEx+, employing multiple Temporal Convolutional Network (TCN) blocks that operate at a fine temporal resolution for both speaker diarization and extraction. Yet, while speech separation and speaker extraction need this fine-grained signal-level output, speaker diarization works at a coarser resolution and doesn't benefit from such precision. Moreover, the Speaker Encoder in USED represents each speaker using a single fixed-dimensional embedding vector that summarizes the entire speaker's characteristics, which limits the representation capacity and poses challenges for the separation

Corresponding author: Quoc Cuong Nguyen.

Ngoc Thuan Tran, and Quoc Cuong Nguyen are with the School of Electrical and Electronic Engineering, Hanoi University of Science and Technology, Hanoi 100000, Vietnam (e-mail: thuan.tn210823@sis.hust.edu.vn; cuong.nguyenquoc@hust.edu.vn).

Ngoc Chau Hoang is with the Viettel AI, Viettel Group, Hanoi 100000, Vietnam (e-mail: chauhn3@viettel.com.vn).

process to effectively utilize the encoded information. Another limitation is that USED requires explicit speaker identity labels for supervising the Speaker Encoder following [21], [22] to learn a highly discriminative embedding to separate speaker identities. While this identity-driven approach ensures a highly discriminative embedding for speaker separation, it inherently prioritizes classification features over the rich, distinctive acoustic properties of speakers - a capability that is essential for effective TSE.

Along with modern diarization approaches come inference pipelines. In [15], Medennikov et al. proposed to estimate target speaker embeddings based on initial diarization results from clustering-based methods before applying TS-VAD. Because TS-VAD requires a fixed maximum number of speakers, [23] introduces a simple strategy for variable speaker scenarios: discard extra speakers if over the limit, and pad with dummy embeddings otherwise. However, selecting an appropriate limit requires prior knowledge of dataset-specific characteristics, which is impractical for real-world inference. In contrast, [24] proposes the *best of both worlds* (BoBW) principle that operates on short chunks with a predefined number of speakers, which is chosen reliably using statistics.

In this work, we propose Speaker-conditioned U-shaped Diarization with Speaker Extraction-guided Enhancement (SUDx), which seamlessly integrates speaker extraction within a framework specifically designed for speaker diarization. Specifically, we propose a speaker-conditioned approach for both speaker diarization and target speaker extraction. By discarding explicit identity supervision, our speaker encoder is optimized to capture distinctive acoustic characteristics specifically to guide the main network. This design choice aligns the speaker encoder's function with that of TSE models [17], [25], prioritizing features most helpful for the extraction process. Next, we adopt the U-net architecture to bridge the resolution gap between diarization and extraction outputs. Our design also incorporates hierarchical speaker representation, which progressively integrates reference features layer-by-layer into mixture representations to disentangle speakers. To handle an unknown number of speakers and overlapping speech, we propose a novel chunk-based inference strategy. Inspired by BoBW principle: by dividing audio into sufficiently small chunks, the maximum number of speakers per chunk can be reliably chosen from data statistics rather than prior knowledge, thereby reducing the need for large pretraining datasets and dataset-specific tuning. For evaluation, the LibriMix dataset [26] is used to assess the effectiveness of SUDx in addressing both speaker diarization and target speaker extraction. To further evaluate diarization performance, we conduct additional experiments on AMI [5], a real-world meeting dataset, under both near-field and far-field settings, as well as on AISHELL-4 [27], a diverse real Mandarin conferencing corpus. Our experimental results show that SUDx achieves state-of-the-art speaker diarization performance even with limited training data, making it highly promising for various real-world applications.

The remainder of this paper is structured as follows. In section II, we review related works on speaker diarization as well as speaker diarization pipelines. Section III describes the proposed methodology in detail. In section IV, we present dataset information and experimental setups. Section V provides experimental results and analysis. Section VI concludes this paper.

II. Related works

A. Speaker Diariziation

Traditional speaker diarization decomposes the task into a sequence of subtasks, such as Speech Activity Detection (SAD), Speech Segmentation, Speaker Embedding Extraction, and Clustering. Most efforts have focused on improving speaker representation and clustering. Various speaker representations have been explored, including i-vector [28], [29], x-vector [30]–[32], and d-vector [2], [33]. For clustering, agglomerative hierarchical clustering (AHC) [1], [11], [34] and spectral clustering [2], [35] are commonly used. By assuming each segment contains only one speaker, traditional methods fail when speakers overlap. Moreover, the cascaded nature of these systems raises the error propagation problem, resulting in suboptimal performance.

To overcome the limitations of traditional diarization systems, EEND approach was proposed [12], [13]. EEND directly predicts frame-level speech activities for each speaker from acoustic features and is trained using permutation-invariant training (PIT) loss [12], [36]. Thanks to the self-attention mechanism, EEND outperforms clustering-based methods and achieves state-of-the-art performance, making self-attention the de facto choice for EEND models. However, early versions of EEND require fixing the number of speakers. To address this, several EEND variants have been developed. SC-EEND [37] decodes speakers one-by-one until a stopping condition is met, while EEND-EDA [14] generates speaker-wise attractors using an LSTM-based encoder-decoder. [38]–[41] continuously developed EEND-EDA by introducing non-autoregressive attractor schemes. While EEND models perform well on short recordings, handling long recordings (e.g., longer than 10 minutes) remains challenging. Processing long sequences as a whole is often impractical due to high memory requirements and limited generalization. A common solution is to split recordings into shorter chunks. In this case, EEND models may suffer from the inter-block label permutation problem, which arises from the PIT training objective treating speaker labels as interchangeable within each chunk. Recognizing the complementarity between clustering-based and EEND methods, Kinoshita et al. [21], [22] proposed EEND-VC, a hybrid approach that applies EEND to short chunks and uses speaker embeddings with clustering to stitch the results. This framework has inspired subsequent models [16], [24], [42], [43], which have further advanced the performance and established themselves as competitive solutions for speaker diarization.

Another effective approach for handling overlapping speech is TS-VAD [15]. Like EEND, TS-VAD performs frame-level prediction of speech activity, but leverages speaker embeddings (e.g., i-vectors, x-vectors) as auxiliary inputs. This design frees TS-VAD from the label permutation problem. However, it requires the number of speakers to be known

in advance. To address this, several extensions have been proposed. Notably, [44], [45] replaces LSTMs with Transformers applied along the speaker axis, allowing the model to process varying numbers of speakers. Nevertheless, the performance of TS-VAD is highly dependent on the quality of speaker embeddings. To mitigate this, recent works such as [46], [47] introduced dictionary learning with more reliable speaker embeddings. Alternatively, [19] integrates TS-VAD with speaker extraction using reference speech instead of speaker embeddings.

On the other hand, several studies have explored the integration of speaker diarization and speech separation [16]–[18]. TS-SEP [18] introduces a two-stage framework in which the model is first pretrained with the TS-VAD objective, followed by finetuning to optimize the separation task. In contrast, EEND-SS [17] proposes a joint end-to-end model that simultaneously optimizes both tasks via multi-task learning. PixIT [16] jointly trains speech separation and speaker diarization through the integration of PIT loss and MixIT loss [48]. More recently, USED [19] explores the integration of speaker diarization and target speaker extraction and found that these two tasks can efficently complement each other. This finding highlights the promising potential of leveraging auxiliary speaker extraction targets to enhance diarization performance. Motivated by this insight, we propose a novel approach that also incorporates speaker extraction, while preserving the modeling characteristics found in EEND and TS-VAD.

B. Inference pipelines for speaker diarization

While clustering-based pipelines are quite straightforward, pipelines of modern approaches, such as EEND or TS-VAD, are more sophisticated. The original TS-VAD pipeline [15] involves multiple steps: initialization with an external diarization systems, followed by several iterations of TS-VAD inference, and final post-processing. This pipeline was further refined in [23] to handle an unknown number of speakers by fixing the number of speakers during training to N. During inference, if the estimated number of speakers $\hat{N} > N$, the least frequent $\hat{N} - N$ speakers are discarded. If $\hat{N} < N$, input is padded with $N - \hat{N}$ i-vectors from the training set. For good performance, N must be chosen sufficiently large and is dependent on the characteristics of the dataset (e.g., $N = 4$ for AMI, $N = 8$ for DIHARD) [23], [46]. This setup requires prior knowledge about the dataset and is impractical for real-world inference. In addition, this also leads to large amounts of pretraining data[1], as pretraining data grows with N. In contrast, the Best-of-Both-Worlds (BoBW) principle [21], [22] used in EEND-VC is more practical. BoBW segments long recordings into shorter chunks, applies EEND to each chunk, and stitches the outputs using speaker embeddings and unsupervised clustering. Bredin et al. [24] further advanced this strategy by using much shorter overlapping windows (i.e., 5s chunk with 500ms step). This design makes it more reliable to set an upper bound on the number of speakers per chunk (e.g., 99% of 5s chunks contain no more than 3 speakers [43]), significantly reducing dependence on extensive pretraining data typically required by fully end-to-end methods [13], [14], [39]–[41]. This inspired us to propose a novel inference strategy, which extends the approach of [23] for our new framework.

[1]Here, "pretraining data" refers to out-of-domain supervised data used to train the model for the intended task (speaker diarization), and does not refer to self-supervised pretraining.

III. Proposed approach

A. Task Definition

Let x(t), with $t \in [0, T_x]$, denote a multi-talker speech mixture where it is assumed that the maximum number of concurrently active speakers is bounded by m. The mixture can be modeled as:

$$x(t) = \sum_{i=1}^{m} c_i(t) + n(t) \tag{1}$$

where $c_i(t)$ represents the clean speech signal corresponding to speaker i, and $n(t)$ denotes additive background noise. If fewer than m speakers are present in the mixture, the corresponding $c_i(t)$ terms are assumed to be zero-valued (i.e., silent).

Let $r_i(t)$, with $t \in [0, T_{r_i}]$ and $i = 1, \ldots, l$ be a set of reference signals for subset of speakers, where $l \leq m$. These references can be obtained either directly via an enrollment phase or indirectly from the non-overlapping regions of the mixture $x(t)$ using an initial diarization step (e.g., clustering-based methods).

We define a joint speaker diarization and extraction system $f(.)$ as follows:

$$(Y, E) = f\left(x, \{r_i\}_{i=1}^{l}\right) \tag{2}$$

where:

- $Y \in \{0, 1\}^{l \times L_{diar}}$ is the frame-wise speech activities, with $Y_{q,t} = 1$ indicating that speaker q is active at diarization frame t, and 0 otherwise. L_{diar} represents the total number of diarization frames.
- $E \in \mathbb{R}^{l \times T_x}$ contains the extracted speech signals, where each row corresponds to the estimated clean speech of a speaker associated with the provided reference signals.

B. System Overview

The overall SUDx model is illustrated in Fig. 1. The input to the framework includes a speech mixture and speech references.

To enable the model to handle a variable number of speakers, we introduce the Speech Assignment Module. All provided references are treated as active speech, while unexpected signals (e.g., silence or speech from non-target speakers) are buffered to accommodate a predefined maximum number of speakers. The weight-shared Speech Encoder transforms both the speech mixture and the selected references into high-resolution, spectrum-like representations. Subsequently, a series of downsampling blocks are employed to reduce the temporal resolution, resulting in higher-level speech feature representations. In parallel, the Multi-scale Feature Extraction Module hierarchically extracts features from the reference signals and integrates them with the mixture representations to progressively support speaker separation.

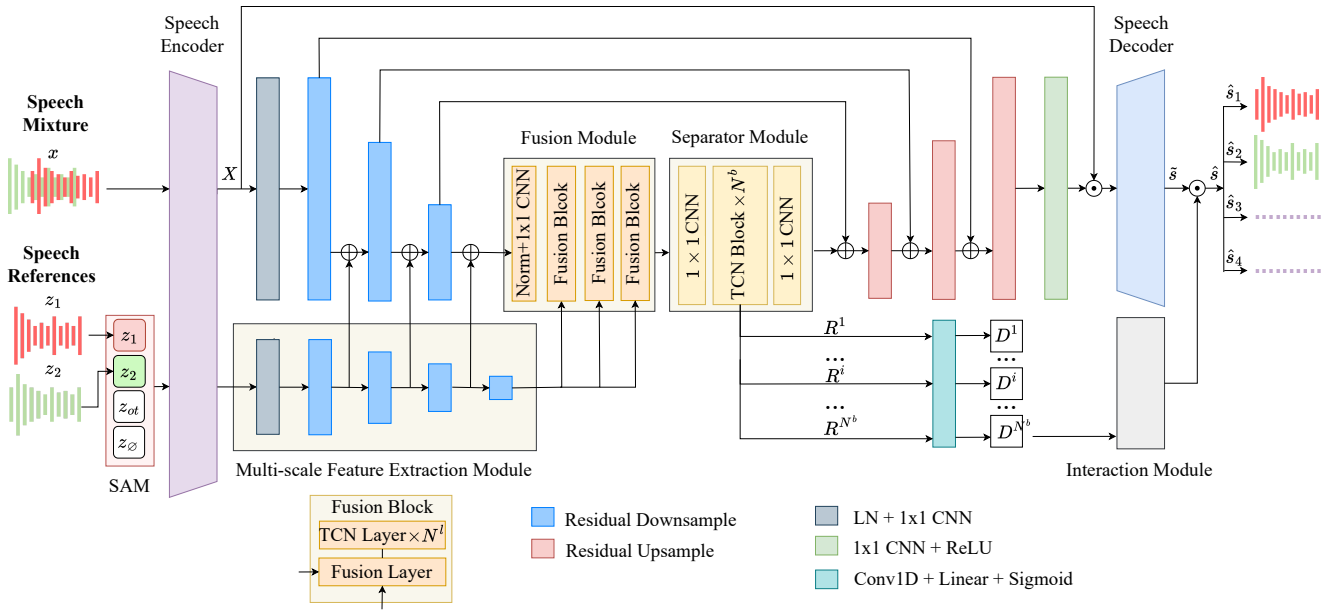


Fig. 1. The overall architecture of SUDx model. The symbols $\oplus$ and $\odot$ are denoted as concatenation and element-wise multiplication, respectively. 1×1 CNN, Conv1D, LN stand for 1D point-wise convolution, 1D convolution and Layer Normalization. SAM is an abbreviation for Speech Assignment Module.

At the bottleneck, the Fusion Module utilizes final speaker embeddings from the Multi-scale Feature Extraction Module for refining the mixture representation into a more discriminative latent space. The Separator Module subsequently operates at this level to isolate individual speaker components within the mixture, producing both intermediate outputs and speaker-specific representations

Speaker speech activity is predicted from the intermediate outputs. Meanwhile, the speaker-specific representations are upsampled through a series of upsampling blocks to estimate separation masks, which are subsequently multiplied with the mixture's spectrum-like features to produce modulated responses. Finally, the Speech Decoder reconstructs the separated speech signals from these modulated responses. The extraction outputs are further refined using diarization information via the Interaction Module, as introduced in [19].

C. Speech Assigment Module

Speech Assignment Module (SAM) enables the model to handle a variable number of speakers and enhances the model's robustness. All provided references are treated as active speeches. To maintain consistency in input dimensionality, SAM assumes a fixed number of speaker slots, denoted by k. If the number of given references is smaller than k, the remaining slots are filled with dummy speech references.

These dummy references are determined based on a decision threshold p_z and fall into one of two categories: (1) learnable zero speech (i.e., a learnable speaker embedding broadcast over time), or (2) speech from speakers not present in the mixture. In addition to the k references, an extra $k+1$-th reference is introduced and always assigned to zero speech. This further strengthens the model's ability to distinguish between active and inactive speakers.

During training, dummy references are supervised to produce silence. To prevent the model from overfitting to specific speaker orders and to improve its robustness, we apply speaker order shuffling across first k references. During inference, if the number of actual speakers is less than k, the remaining slots are automatically filled with zero speech. The details of SAM are shown in Algorithm 1.

D. Multi-scale Feature Extraction Module

The Multi-scale Feature Extraction Module is designed to capture speaker-related information across multiple temporal resolutions, which are combined with hierarchical representations of the mixture to progressively separate speakers. The core of this module is a series of downsampling blocks.

Downsampling and upsampling are integral components of the U-net based architecture. Through empirical observation, we found that, in addition to the Fusion Module and Separator Module, these components have substantial impact on overall system performance. Therefore, we adopted the Residual Downsampling and Residual Upsampling concept from [49], which achieved remarkable performance on the speech enhancement task (see Fig. 2). In this design, dilated convolutions are employed to increase the receptive field, while residual connections help mitigate information loss during temporal compression.

Our system incorporates two downsampling branches: one for the noisy mixture and one for the reference signals (Multi-scale Feature Extraction Module). This design serves two main purposes. First, it aims to extract multi-level features from the speech references, which are then combined with the downsampled mixture representations to separate speaker progressively. This hierarchical speaker representation has been validated for effective detaching speakers and become a common trend in recent state-of-the-art TSE models [50]–

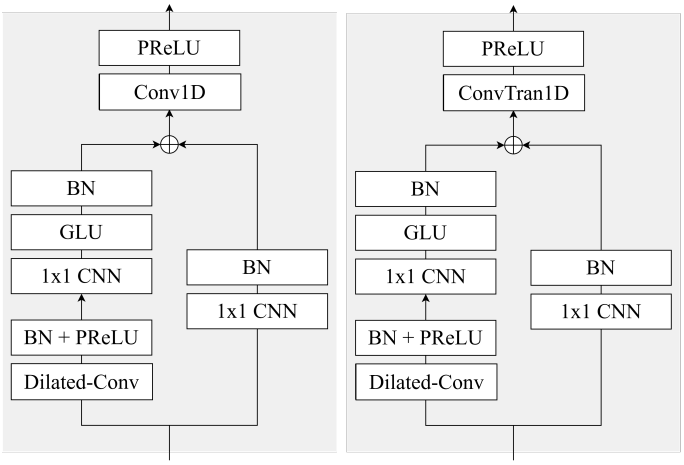
Algorithm 1: Speech Assignment Algorithm

Data:
k: Number of speaker slots
$\mathcal{S}_{all}$: Set of all speakers
$\mathcal{S}_{enroll}$: Set of enrollment speakers
$z_{\varnothing}$: Learnable zero speech
p_z: Probability to use zero speech as input
// List of active, dummy speech signals
$I_{act}, I_{dum} \leftarrow [\,], [\,]$;
// Number of input speech signals
$count \leftarrow 0$;
/* Accept all given references as active speech signals */
for s *in* $\mathcal{S}_{enroll}$ **do**
  I_{act}.add(z_s);
  $count \leftarrow count + 1$;
end
/* Adding dummpy speech references */
while $count < k$ **do**
  if *is training* **then**
    $p \sim \mathcal{U}[0, 1]$;
  else
    $p \leftarrow 0$;
  end
  if $p > p_z$ **then**
    // Randomly select a speaker not in the current mixture
    $s_{\text{ot}} \sim \mathcal{U}[\mathcal{S}_{\text{all}} \setminus \mathcal{S}_{\text{enroll}}]$;
    I_{dum}.add($z_{s_{ot}}$);
  else
    I_{dum}.add($z_{\varnothing}$);
  end
  $count \leftarrow count + 1$;
end
I_{dum}.add($z_{\varnothing}$);
Output: I_{act}, I_{dum}

[52]. By capturing features across multiple scales, the model can encode speaker characteristics more diversely and efficiently than relying solely on a single representation. Second, our design serves to reduce the temporal resolution, thereby yielding more compact and abstract representations. By lowering temporal granularity, we emphasize high-level, speaker-specific features, rather than fine-grained, transient details. These lower-resolution representations are well-suited for speaker diarization, as the task prioritizes distinguishing consistent speaker identities over precise temporal accuracy. Before downsampling, the spectrum-like features from the Speech Encoder are first passed through Layer Normalization, followed by a point-wise convolutional (1×1 CNN) layer.

The multi-scale speaker features of the Multi-scale Feature Extraction Module are obtained by applying average pooling along the time dimension to the outputs of the downsampling blocks. These embeddings are then concatenated with mixture representations at the corresponding levels, which we name as Pyramidal Fusion. To obtain more discriminative speaker embeddings, we further apply an additional downsampling block, followed by a 1×1 CNN layer, on top of the previous reference representations. These final speaker embeddings are used as auxiliary inputs for the Fusion Module.

The high-level representations of mixture are fed into bottleneck modules (i.e., the Fusion Module and Separator Module). The output of the bottleneck stage is then upsampled to restore the temporal resolution that was reduced during the downsampling process. The residual upsampling block closely mirrors the downsampling block, except the replacement of the final one-dimensional convolutional (Conv1D) layer by a one-dimensional transposed convolutional (ConvTran1D) layer. Before each upsampling block, the input is formed by concatenating the cached representation from the corresponding downsampling block with the output of the previous block.



(a) Residual Downsample (b) Residual Upsample

Fig. 2. The internal structure of Residual Downsample and Upsample. BN, PReLU, Dilated-Conv are batch normalization, parametric ReLU, and dilated convolution. The symbol $\oplus$ denotes element-wise addition.

E. Fusion Module

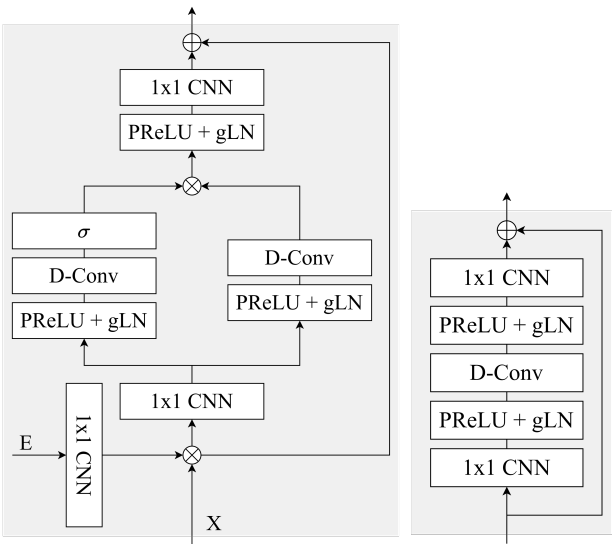
The Fusion Module is designed to integrate the final speaker features from the Multi-scale Feature Extraction Module with high-level feature representation of the speech mixture. This integration plays a critical role in guiding the model to effectively separate individual speakers.

The mixture representation first passes through a Layer Normalization step, followed by a 1×1 CNN layer. The result then is fused with the input speaker embeddings and processed by a stack of N_b Fusion blocks. Each Fusion block is composed of a Fusion Layer and a subsequent stack of N_l TCN Layers. Because the integration happens in the bottleneck, we name this process Bottleneck Fusion.

Concatenation is a commonly adopted fusion strategy, as seen in prior works such as [20], [53]. However, [54] suggests that simple concatenation may lead to information overwhelming, thereby degrading performance. In our implementation, we adopt the fusion mechanism proposed in [52]. The structure of the Fusion layer is illustrated in Fig. 3a.

TCNs are widely employed in speech separation and speaker extraction tasks [20], [55] due to their efficiency, lower computational complexity, minimal memory requirements, faster in training and inference.

In our design, the TCN layers further refine the output of the Fusion Layer, enhancing speaker separation. The structure of TCN Layer is shown in Fig. 3b. We stack N_l TCN layers with exponential increasing dilation factor of 2^n, where $n \in \{1, ..., N_l\}$.



(a) Fusion Layer (b) TCN Layer

Fig. 3. The internal structure of Fusion layer and TCN layer. PReLU, gLN and D-Conv are parametric ReLU, global layer normalization and dilated depth-wise separable convolution. E denotes the speaker representation, and X represents the mixture representation. The symbols $\oplus$ and $\otimes$ denote element-wise addition and element-wise multiplication.

F. Separator Module

The Separator Module is responsible for isolating individual speakers based on the output produced by the Fusion Module. Its function is analogous to the separator modules used in conventional speech separation systems [55]–[57], which aim to isolate speakers using learned basis representations.

In our approach, the fused output from the Fusion Module is first passed through a 1×1 CNN layer. This layer concatenates $k+1$ representations along the channel dimension and projects it into a latent space of basis functions. Following this, we apply a stack of N_b TCN Block, each composed of N_l+1 TCN Layer with increasing dilation factors of 2^n, where $n \in \{0, ..., N_l\}$. These stacked blocks iteratively refine the representation, producing intermediate outputs $\{R_1, ..., R_{N_b}\}$. These intermediate representations serve as input for predicting speaker speech activity. Finally, the output of the last TCN block, R_{N_b}, is decoded via a 1×1 CNN layer into $k+1$ separate streams for each speaker. These decoded streams are subsequently upsampled to restore the temporal resolution before being used to estimate the clean speech signals.

G. Loss Function

The diarization probability outputs are denoted as $D^j = \{D_1^j, ..., D_{k+1}^j\}$, $j = 1, ..., N_b$, and the final extracted signals are represented as $\hat{s} = \{\hat{s}_1, ..., \hat{s}_{k+1}\}$, where each $\hat{s}_i = \{\hat{s}_i^1, \hat{s}_i^2, \hat{s}_i^3\}$ corresponds to the three time scales produced by the Speech Decoder.

At inference time, only the diarization output from the final block and only the first component of each final signals, $\hat{s}^1 = \{\hat{s}_1^1, ..., \hat{s}_{k+1}^1\}$, are utilized for prediction. The diarization probability outputs from the final block, D^{N_b}, are subsequently thresholded to obtain the frame-wise speech activity matrix Y.

1) Overall Loss Function: The total loss function comprises the speaker diarization loss and the speaker extraction loss, formulated as:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{ext} + \lambda_2 \mathcal{L}_{diar} \tag{3}$$

where λ_1 and λ_2 are weighting coefficients for the speaker extraction loss and speaker diarization loss.

2) Speaker Extraction: We categorize training scenarios into four distinct classes, following the taxonomy proposed in [58]: QQ (*quiet-quiet*), QS (*quiet-speaking*), SS (*speaking-speaking*), and SQ (*speaking-quiet*). In these labels, the first letter corresponds to the target speaker and the second to the interfering speaker.

When the target speaker is quiet, the system minimizes the SAD loss [58], defined as:

$$\mathcal{L}_{\mathrm{E}} = \sum_{j=1}^{3} 10\mu_j \log_{10}\left(\frac{\|\hat{s}^j\|^2}{T_{\mathrm{se}}} + \epsilon\right) \tag{4}$$

where T_{se} is the duration of $\hat{s}^j$ in seconds, μ_j is the weight of the j-th ConvTran1D output, and ϵ is a small constant for numerical stability, set to 10^{-6}.

Conversely, when the target speaker is speaking, the SI-SDR loss [59] is used to evaluate the discrepancy between the ground-truth speech s and extracted speech $\hat{s}$:

$$\begin{cases} s_{target}^j = \frac{\langle \hat{s}^j, s\rangle s}{\|s\|^2+\epsilon} \\ e_{noise}^j = \hat{s}^j - s_{target}^j \\ \mathcal{L}_S = -10\sum_{j=1}^{3}\mu_j \log_{10}\frac{\left\|s_{target}^j\right\|^2}{\left\|e_{noise}^j\right\|^2+\epsilon} + \epsilon \end{cases} \tag{5}$$

Finally, the total speaker extraction loss is computed by aggregating losses across all $k+1$ outputs, weighted by scenario-specific coefficients (α, β, γ, and δ):

$$\mathcal{L}_{ext} = \frac{1}{k+1}\left(\sum_{i=1}^{k+1} \alpha\mathcal{L}_{E_i}^{QQ} + \beta\mathcal{L}_{E_i}^{QS} + \gamma\mathcal{L}_{S_i}^{SS} + \delta\mathcal{L}_{S_i}^{SQ}\right) \tag{6}$$

3) Speaker Diarization: The binary cross-entropy (BCE) loss has been demonstrated to be effective for training speaker diarization systems, as shown in [15], [46]. We denote $\mathcal{L}_{diar}^j$ as the loss associated with the j-th output of the Diarization Decoder, which is computed as:

$$\mathcal{L}_{diar}^j = \frac{1}{k+1}\sum_{i=1}^{k+1} \mathrm{BCE}\left(y_i^{diar}, D_i^j\right) \tag{7}$$

where y_i^{diar} represents the ground-truth speech activity label for the i-th output.

The overall diarization loss is then obtained by summing over all N_b decoder outputs:

$$\mathcal{L}_{diar} = \sum_{j=1}^{N_b} \mathcal{L}_{diar}^j \tag{8}$$

H. Proposed Inference Pipeline

Inspired by the BoBW concept, we introduce a novel, simple, and effective inference strategy with a similar purpose. Unlike the original BoBW [22], [24], which was proposed for EEND systems and follows a multi-stage process of local diarization followed by global clustering, our pipeline adopts the opposite order—clustering first, then diarization—following the principles of [15], [23] and is applicable to TS-VAD systems. However, compared with the current inference pipelines, which are highly dataset-dependent and require large amounts of pretraining data, our approach incorporates several enhancements, including the use of small chunk sizes borrowed from BoBW, to reduce dataset dependency and alleviate the need for extensive pretraining. The strategy proceeds as follows:

1) **Initial Diarization**: Generate initial diarization results using an external system (e.g., clustering-based methods)
2) **Embeddings/References Extraction**: Base on the estimated speaker activity, extract speaker embeddings or speech reference for each predicted speaker
3) **Chunking**: Segment the conversation into overlapping chunks using a sliding window of size L_w and step size L_s. The chunking strategy should ensure that most chunks contain no more than the assumed maximum number of speakers (e.g., 99% of 5-second chunks contain no more than 3 speakers [43])
4) **Chunk Filtering and Merging**: Discard any chunks that contain no active speakers. Merge adjacent chunks when their combined speaker count does not exceed the maximum allowed. Limit the total duration of merged chunks using a hyperparameter L_c
5) **Local Diarization Inference**: Apply model based TS-VAD on each valid chunk
6) **Final aggregation and Post-processing**: Stitch the chunk-level results into a continuous diarization output. Apply post-processing techniques such as filtering, deletion of short segments shorter than δ, and concatenation of short pauses shorter than Δ

IV. Experimetal setup

A. Dataset

We conduct evaluations of our model on two benchmark datasets. To simultaneously assess speaker diarization and speaker extraction, we use LibriMix, which is simulated. For validating the effectiveness of the proposed network and our novel inference strategy, we use AMI, a real meeting corpus.

1) LibriMix: The LibriMix dataset [26] is a publicly available benchmark designed to support a variety of speech processing tasks, including speech separation, speaker extraction, and speaker diarization. It is constructed by combining clean speech segments from the LibriSpeech corpus with background noise from the WHAM! dataset, all resampled to 16 kHz. LibriMix includes two main configurations: Libri2Mix and Libri3Mix. The Libri2Mix 100h and Libri3Mix 100h subsets are derived from the train-clean-100 and test-clean portions of LibriSpeech, containing 58h/11h/11h (2 speakers) and 40h/11h/11h (3 speakers) for training, validation, and testing, respectively.

Two mixing modes are provided: *min mode* (ending at the shortest utterance) and *max mode* (padding shorter utterances to match the longest). We conducted our experiments using the max mode, as it is more suitable for both speaker diarization and speaker extraction tasks. To ensure fair comparison with previous studies and maintain compatibility with the speaker extraction task, we adopted the data split [2] used in [19].

2) AMI: To evaluate our model under realistic conversational conditions, we utilize the AMI Meeting Corpus [5], a well-established benchmark in the domain of speaker diarization and multimodal meeting analysis. AMI captures the acoustic and linguistic variability typical of everyday dialogue, where overlapping speech, room acoustics, and non-native accents challenge the limits of diarization systems. The corpus spans approximately 100 hours of multi-party meeting recordings across 171 sessions, each involving 3 to 5 speakers. AMI provides both close-talking audio, captured using lapel and headset microphones, and far-field audio, recorded via tabletop and ceiling-mounted microphone arrays. Two famous variants of the AMI corpus are IHM (Individual Headset Microphone) and SDM (Single Distant Microphone).

In this study, we followed the data split and annotation protocols based on the full-corpus ASR partition [3]. This split divides AMI into training, validation, and test sets with approximate durations of 80 hours, 10 hours, and 9 hours, respectively. Our proposed model is first pretrained on the LibriMix, as introduced in the previous subsection, and is then fine-tuned on the AMI training set. During inference, we follow our proposed principle. Speech references can be extracted based on the initial diarization results produced by a clustering-based model. This initial diarization is obtained through the combination of a pretrained ECAPA-TDNN [60], trained on the VoxCeleb2 dataset [61] [4], and the Spectral Clustering algorithm implemented in the SpeechBrain toolkit [62].

Although the AMI corpus includes headset recordings that could be used for speech separation, in this work, we only employ the AMI diarization annotations for fine-tuning our model.

3) AISHELL-4: To further evaluate our system in complex, real-world conversational environments, we utilize the AISHELL-4 dataset [27], a sizable, real-recorded Mandarin speech corpus collected for the conference scenario. The corpus consists of 120 hours of recorded meeting sessions (211 sessions total), with each session involving 4 to 8 participants. Data collection was performed across 10 real conference venues of varying sizes—categorized as small, medium, and large—using an 8-channel circular microphone array. AISHELL-4 provides realistic acoustics and rich natural speech characteristics, including noise, quick speaker turns, and a noticeable amount of speech overlap.

In this work, we adopt the provided training and evaluation split. To obtain a single-channel input, all microphone

[2] https://github.com/msinanyildirim/USED-splits
[3] https://github.com/BUTSpeechFIT/AMI-diarization-setup
[4] https://github.com/TaoRuijie/ECAPA-TDNN

channels are mixed into one channel. The overall training and inference pipeline follows the same procedure as described for the AMI corpus.

B. SUDx Model Configuration

For evaluation on both the LibriMix dataset and the AMI corpus, the maximum number of speakers k is set to 3. The Speech Encoder and Decoder are designed following a multi-scale architecture [20]. Specifically, three parallel encoder–decoder branches are employed, each configured with 256 channels, kernel sizes of 20, 80, and 160, respectively, and a uniform stride of 10. The mixture branch consists of 3 downsampling blocks, with a symmetric number of upsampling blocks in the decoder. The Multi-scale Feature Extraction module follows the same architectural design as the mixture branch, with an additional downsampling block. Both downsampling and upsampling blocks use the same kernel size of 3. The final convolutional layer applies a stride of 2, halving the temporal resolution at each step. The dimensionality of the final speaker embeddings of Multi-scale Extraction Module is set to 256.

For the TCN, we configure N_l as 7 and the number of blocks N_b as 3. The convolutional layer used to generate diarization outputs employs a kernel size of 4 and a stride of 2, with both input and output channel dimensions set to 256. This channel dimension is also shared by the three parallel 1×1 CNN layers used to generate the separation masks. Additionally, the convolutional layer in the multi-task interaction module uses a kernel size of 3.

We follow prior studies [20], [53] in assigning the weights for the multi-scale SI-SDR loss, setting $\mu_1 = 0.8$, $\mu_2 = 0.1$, and $\mu_3 = 0.1$ for the high, middle, and low temporal resolutions, respectively, as it was reported that the highest temporal resolution yields the best performance. The loss weighting coefficients α and β are both set to 0.001, while γ and δ are set to 1.0. The overall loss is composed of speaker extraction and diarization losses, with equal weighting $\lambda_1 = \lambda_2 = 1.0$. The probability to use zero speech as input p_z is set to 0.3.

Training is performed using the Adam optimizer for 11 epochs on a single NVIDIA P100 GPU. Input utterances are segmented into 4-second chunks with a 2-second shift. The training batch size is 4. The learning rate is set to 10^{-4}, which is warmed up for the first 10% steps, and then decayed using a polynomial schedule. For domain adaptation, the model is fine-tuned on in-domain data using a fixed learning rate of 10^{-5} to reduce the condition mismatch. The overall loss on the validation set is used as the primary metric for model checkpoint selection.

C. Evaluation metrics

We use Diarization Error Rate (DER (%)), including Speaker Confusion (SC (%)), False Alarm (FA (%)), and Missed Detection (MS (%)) to evaluate the Speaker Diarization performance. Forgiveness collar and median filtering are set to 0 seconds and 11 frames for both LibriMix and AMI corpus. We also assess the Scale-Invariant Signal-to-Distortion Ratio improvement (SI-SDRi (dB)) for speaker extraction performance. Higher SI-SDRi values indicate better source reconstruction quality, while lower values of DER, SC, FA, and MS reflect more accurate diarization performance.

V. Experimental results and analysis

A. Ablation studies

We conduct ablation studies to evaluate the contribution of several key components to the overall performance of the model, as presented in Table I. First, we add a speaker loss to supervise the learning of global embeddings, a strategy commonly adopted in TSE models [20], [63]. Here, we add a Linear layer on top of the final speaker embeddings from the Multi-scale Feature Extraction Module to map the speaker embedding dimension D_{spk} to the number of speakers in the training set, then we use the Cross-Entropy loss with the weight of 1 for supervision. Notably, adding the speaker loss leads to a clear degradation in performance, both in diarization (DER increases from 4.46% to 5.28%) and in source extraction (SI-SDRi drops from 11.94dB to 11.42dB). This reduction in performance can be attributed to the speaker loss, causing the model to focus too much on speaker identity, potentially at the expense of learning semantically meaningful representations. These results suggest that the speaker loss may distract the model from capturing the relevant speaker characteristics needed for both diarization and source extraction. As a result, the number of speaker confusion errors increases (0.18% to 0.25%), which negatively impacts performance across other metrics. The hierarchical speaker representation also plays a crucial role in the overall performance of the model. Their multi-scale nature allows the network to progressively separate speakers at different temporal resolutions, which facilitates more effective detaching speakers compared to relying solely on global features. Without other features (i.e., removing Pyramidal Fusion), the performance noticeably degrade (DER rises from 4.46% to 4.98%, and SI-SDRi drops from 11.94dB to 11.57dB). Lastly, we investigate the effect of the U-shaped architecture by removing both the downsampling and upsampling blocks from the model. This results in a notable increase in diarization error, while the source extraction performance appears to improve slightly. We are going to investigate this phenomenon further in the subsequent subsection.

TABLE I
Ablation study results on the LibriMix dataset max mode. The best performance is highlighted in bold.

Model	DER↓	MS+FA↓	SC↓	SI-SDRi↑
SUDx	**4.46**	**4.28**	**0.18**	11.94
+ Speaker Loss	5.28	5.03	0.25	11.42
- Pyramidal Fusion	4.98	4.79	0.19	11.57
- Downsample	5.50	5.15	0.35	**11.95**

B. Impact of the U-shaped architecture

To further investigate the impact of the U-shaped architecture, we conduct a series of experiments by varying the number of residual downsampling and upsampling blocks. To isolate

the effect of the architecture itself, we disable the Pyramidal Fusion. The same Multi-scale Feature Extraction Module is used, with the number of downsampling blocks fixed at four, and all other configurations follow those described in Section IV-B. To assist the global features in distinguishing between speakers and maintain a consistent representation space across experiments, we also incorporate the speaker loss with the weight of 0.1. The results of these experiments are presented in Table II.

As shown in Table II, the diarization performance generally improves as the number of downsampling blocks increases, except for the case of 2 downsampling blocks. While the extraction quality shows the opposite trend, highlighting the importance of maintaining high temporal resolution. Indeed, higher temporal resolution allows the model to capture fast changes in the speech signal, which are crucial for effective separation. When downsampling is applied, short-term features may be lost, leading to reduced extraction performance. In contrast, precise temporal resolution is less critical for the diarization task, as the goal is to determine who spoke when, rather than what was said. Consequently, the U-shaped architecture becomes more suitable, as it allows the integration of broader contextual information across time. Nonetheless, the table also suggests that the number of downsampling blocks must be appropriate to balance temporal resolution and representational abstraction. Otherwise, it leads to suboptimal results for both tasks, as seen in the case of two downsampling blocks.

TABLE II
MODEL PERFORMANCE ON THE LIBRIMIX DATASET MAX MODE WHEN SETTING DIFFERENT NUMBER OF DOWNSAMPLING BLOCKS.

N.o. Downsample	DER↓	MS+FA↓	SC↓	SI-SDRi↑
1	5.26	5.00	0.26	**12.19**
2	5.33	5.08	0.25	11.76
3	4.89	4.63	0.25	11.69
4	**4.83**	**4.59**	**0.24**	11.69

C. Impact of the hierarchical speaker representation

As mentioned, SUDx integrates multi-scale speaker features with mixture representations via two fusion mechanisms: Pyramidal Fusion, which operates on embeddings during the downsampling process, and Bottleneck Fusion, which utilizes the final speaker embeddings from the Multi-Scale Feature Extraction Module. To evaluate the contribution of each, we isolate and assess them independently, as shown in Table III. The results indicate that both fusion strategies play an important role, with Pyramidal Fusion having a more pronounced impact. When combining them, diarization accuracy improves, while extraction quality slightly declines. The decrease of DER indicates the effectiveness of integration speaker embeddings at multi levels. Meanwhile, the slight decrease in SI-SDRi is also explainable: when using only Pyramidal Fusion, the Fusion Module indirectly acts as an additional Separator Module, thereby enhancing the separation quality.

TABLE III
IMPACT OF DIFFERENT FUSION MECHANISMS ON THE LIBRIMIX DATASET MAX MODE.

Fusion	DER↓	MS+FA↓	SC↓	SI-SDRi↑
Pyramidal	4.53	4.34	0.19	**12.08**
Bottleneck	4.98	4.79	0.19	11.57
Pyramidal + Bottleneck	**4.46**	**4.28**	**0.18**	11.94

D. Comparison with other state-of-the-art methods

1) Results on LibriMix: To confirm the effectiveness of our proposed architecture on both diarization and speaker extraction tasks, we compare its performance with several state-of-the-art systems on the LibriMix dataset in the *max mode* setting, which is identical experimental conditions, following the same evaluation protocol and baseline setup as [19]. The baselines include HuBERT BASE [65] and wav2vec 2.0 BASE [64], two self-supervised learning models from the SUPERB diarization task [66], TS-VAD [15] and SpeX+ [20], representing strong baselines for speaker diarization and extraction, respectively. In addition, USED-F [19], which has demonstrated superior performance across both tasks, is also included for comparison. Table IV presents our comparison using DER, including MS, FA, and SC, along with the SI-SDRi metric.

As can be seen, our proposed model outperforms almost all baseline methods across the evaluation metrics, except for the SI-SDRi score (11.94dB vs. 12.70dB of USED-F) and the FA score (2.18% vs. 2.16 of USED-F). Because FA is quite correlated with MS, these numbers usually change in two different directions when the threshold changes, we used the sum FA and MS instead. We can see that our model slightly outperforms USED-F in MS+FA. A notable result is the SC value, which has nearly 50% relative improvement over the lowest. The lower SC indicates that the model is more effective at distinguishing between speakers, which provides a strong foundation for potential improvements in speaker extraction quality. This phenomenon demonstrates the effectiveness of using the U-shaped architecture and the hierarchical speaker representation for the diarization task.

We also compare our model with USED-F on the speaker diarization task in isolation by disabling the extraction module (i.e., setting λ_1 to 0). The results are reported in Table V. Our model continuously outperforms USED-F by a significant margin even without the support of the extraction module. These findings further validate the effectiveness of our architectural design for speaker diarization. In addition, the performance difference between the full model and its diarization-only variant highlights the pivotal role of the extraction module in supporting the diarization task.

2) Results on AMI-IHM: For real data, unlike [19], which was evaluated on the CALLHOME [67] telephone dataset, we opt for AMI corpus, a realistic and challenging meeting corpus. Our baselines include traditional clustering-based model, VBx [40], fully end-to-end models like EEND [68], EEND-EDA [68], DiaPer [40], and hybrid approaches from the PyAnnote series [24], [43]. For clarity, information not

TABLE IV
COMPARISON WITH STATE-OF-THE-ARTS ON THE LIBRIMIX DATASET MAX MODE. THE RESULTS OF BASELINE SYSTEMS ARE TAKEN FROM [19]. "PARAMS" REFERS TO THE NUMBER OF PARAMETERS COUNTED IN MILIONS (M).

Model	Params	DER↓	MS+FA↓	MS↓	FA↓	SC↓	SI-SDRi↑
wav2vec 2.0 BASE [64]	95	7.62	7.10	2.28	4.82	0.52	N/A
HuBERT BASE [65]	95	7.56	7.21	2.40	4.81	0.35	N/A
TS-VAD [15]	39.50	7.28	6.39	3.61	2.78	0.89	N/A
SpEx+ [20]	16.35	N/A	N/A	N/A	N/A	N/A	9.06
USED-F [19]	23.12	4.75	4.34	2.18	**2.16**	0.42	**12.70**
SUDx	20.83	**4.46**	**4.28**	**2.10**	2.18	**0.18**	11.94

TABLE V
COMPARISON WITH USED-F [19] FOR THE SPEAKER DIARIZATION TASK ONLY (ONLY SD) ON THE LIBRIMIX DATASET MAX MODE. THE UNDERLINE INDICATES THE BEST DER AMONG ONLY SD MODELS.

Model	DER↓	MS↓	FA↓	SC↓
USED-F	4.75	2.18	**2.16**	0.42
USED-F (Only SD)	6.43	-	-	-
SUDx	**4.46**	**2.10**	2.18	**0.18**
SUDx (Only SD)	<u>5.63</u>	2.57	2.80	0.26

explicitly reported in the original papers is omitted. Among these baselines, the PyAnnote models are conceptually closest to our work because of the BoBW principle. For further demonstration of the effectiveness of our model, we modify our network and create a USED-F-like model by simply removing the downsample and upsample blocks, then replacing the Multi-scale Feature Extractor Module with a speaker encoder consisting of several ResNet blocks, as described in [19]. We also apply our inference strategy to USED-F during evaluation. Experiments are shown in Table VI.

We start by using ECAPA-TDN and Spectral Clustering as our initial diarization results. We can see very clearly that the traditional approach performs poorly on corpora with high levels of overlapping speech, yielding a large number of FA+MS. This underscores the need for an effective refinement strategy and a high-performance model for post-processing. To address this, we applied our inference strategy to refine the initial diarization outputs. As can be seen, adding refinement network on top of the initial results greatly boosts performance (from 31.83% to 21.09% and 18.13% in USED-F-like and SUDx, respectively). We can see that USED-F-like surpasses VBx and all fully end-to-end methods. However, when comparing with PyAnnote 2.1 and 3.1, USED-F-like still lags behind. While our model already demonstrated superior speaker separation performance on the LibriMix simulated dataset (SC of 0.18% vs. 0.42% for USED-F), the gap widens further on real conversational data (3.78% vs. 5.64%). This result highlights the effectiveness of the hierarchical speaker representation enables robust speaker assignment even with degraded or imperfect reference signals. Our proposed model can easily outperform nearly all baselines, except for the PyAnnote 3.1 with competitive performance (18.13% vs. 18.00%). It is important to emphasize that our model was trained with over 100 times less simulation data compared to full end-to-end approaches (98h vs. more than 10,000h), yet it still significantly overshadows these methods (18.13% vs. 21.56% of EEND-EDA). The PyAnnote series is well known for their robustness, thanks to extensive pretraining on a real compound dataset (e.g., DIHARD [69], and AMI [5]). Despite using 3-5 times less pretrained data and relying solely on simulation data, our SUDx still considerably outperforms PyAnnote 2.1 and achieves comparable performance to PyAnnote 3.1. These results confirm the effectiveness of the proposed U-shaped architecture, hierarchical speaker representation, and inference strategy. Together, they enable robust and accurate speaker diarization in acoustically challenging conditions, achieving state-of-the-art performance comparable to BoBW-based methods.

3) Results on AMI-SDM: To evaluate model performance under far-field conditions, we further conduct experiments on the AMI-SDM variant, as reported in Table VII. The proposed SUDx consistently outperforms both traditional clustering-based and fully end-to-end diarization models. However, SUDx slightly underperforms the PyAnnote models, with a small DER gap (25.09% vs. 22.9% for PyAnnote 3.1). It is worth noting that SUDx is pretrained on a relatively small amount of simulated data, whereas the PyAnnote series is trained on larger compound of real-world datasets that also include far-field data (e.g., AISHELL-4). Despite this data disparity, SUDx achieves the lowest speaker confusion among all systems, which aligns with its consistent performance trends observed on both LibriMix and AMI datasets. These results highlight the robustness and generalization capability of the proposed speaker-conditioned framework, demonstrating its ability to effectively disentangle speakers even under acoustically adverse far-field scenarios.

4) Results on AISHELL-4: The AISHELL dataset introduces additional challenges for evaluating the robustness of the proposed SUDx, including differences in language, a larger number of speakers per session, and more diverse acoustic conditions. The comparison results are summarized in Table VIII. A trend similar to that observed on the AMI-IHM set can be seen: SUDx consistently outperforms almost all methods, including PyAnnote 2.1, and achieves competitive performance with PyAnnote 3.1 (13.53% vs. 13.20%), while demonstrating superior speaker separation capability (3.47% vs. 6.60%). Remarkably, although SUDx is pretrained on only a small amount of simulated English data and employs an

TABLE VI
COMPARISON WITH STATE-OF-THE-ARTS ON THE AMI HEADSET MIX (IHM) DATASET. SIM AND COM STAND FOR SIMULATION AND COMPOUND RESPECTIVELY. SC* DENOTES SPECTRAL CLUSTERING TO AVOID CONFLICT WITH SC (SPEAKER CONFUSION).

Model	Pretraining Data	Data type	DER↓	MS+FA↓	SC↓
VAD+VBx+OSD [40]	-	N/A	22.42	-	-
EEND [68]	>10,000h	sim	27.70	-	-
EEND-EDA [68]	>10,000h	sim	21.56	-	-
DiaPer [40]	>10,000h	sim	30.49	-	-
PyAnnote 2.1 [24]	249h	com	18.50	14.00	4.40
PyAnnote 3.1 [43]	509h	com	**18.00**	-	-
ECAPA-TDNN + SC*	-	N/A	31.83	29.18	2.55
+ USED-F-like	**98h**	sim	21.09	15.45	5.64
+ SUDx	**98h**	sim	18.13	14.35	3.78

TABLE VII
PERFORMANCE COMPARISON AGAINST STATE-OF-THE-ART SYSTEMS ON THE AMI-SDM DATASET. PRETRAINING DATA AND DATA TYPES ALREADY PRESENTED IN TABLE VI ARE OMITTED HERE FOR BREVITY.

Model	DER↓	MS+FA↓	SC↓
VAD+VBx+OSD [40]	34.61	-	-
DiaPer [40]	50.97	-	-
PyAnnote 2.1 [24]	**22.20**	16.00	6.20
PyAnnote 3.1 [43]	22.90	-	-
SUDx	25.09	20.17	**4.92**

TABLE VIII
EVALUATION RESULTS ON THE AISHELL-4 DATASET AGAINST STATE-OF-THE-ART APPROACHES. PRETRAINING DATA AND DATA TYPES ALREADY PRESENTED IN TABLE VI ARE OMITTED HERE FOR BREVITY.

Model	DER↓	MS+FA↓	SC↓
VAD+VBx+OSD [40]	15.84	-	-
DiaPer [40]	31.30	-	-
PyAnnote 2.1 [24]	14.50	7.90	6.60
PyAnnote 3.1 [43]	**13.20**	-	-
SUDx	13.53	10.06	**3.47**

English speaker embedding model, it generalizes effectively to Mandarin speech. These findings further validate the effectiveness of the proposed U-shaped architecture and hierarchical speaker representations, demonstrating both robustness and strong generalization across linguistically and acoustically challenging far-field scenarios. In addition, the results also underscore the practicality of our inference strategy for handling an unknown number of speakers: even when trained with only 2–3 speakers, the model performs reliably on sessions with up to 8 speakers, achieving state-of-the-art performance.

E. Novel inference strategy analysis

To further evaluate the effectiveness of the proposed pipeline, we conduct an analysis, the results of which are presented in Table IX. The pipeline introduced in [23] is used as our baseline. Specifically, input recordings are chopped into 15s chunks, and SUDx is then applied to each chunk. The main differences of the proposed pipeline from baseline are shorter chunk size (i.e., 4s) and the inclusion of a merging step. If the merging step in the proposed strategy is excluded, our pipeline closely resembles that of the baseline.

TABLE IX
PERFORMANCE COMPARISON AND ABLATION ANALYSIS OF THE PROPOSED INFERENCE STRATEGY ON THE AMI HEADSET MIX DATASET.

Pipeline	DER↓	MS+FA↓	SC↓
Pipeline of [23]	18.92	14.60	4.32
Proposed (w/o Merging)	20.03	16.85	**3.18**
Proposed	**18.13**	**14.35**	3.78

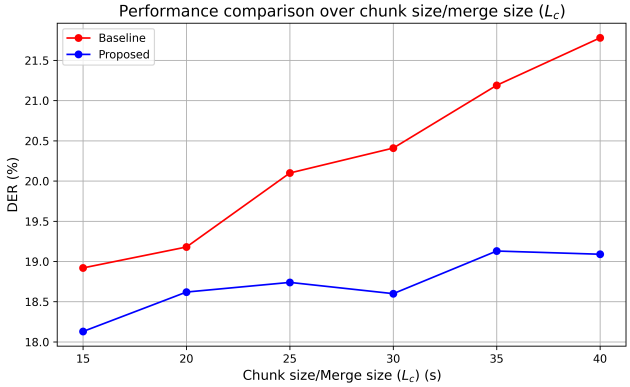


Fig. 4. Performance comparison between the baseline [23] (red) with varying chunk size and the proposed pipeline (blue) with varying merge size (L_c) on the AMI Headset Mix dataset.

We first compare the baseline configuration, which uses a longer chunk size, with the proposed strategy that employs shorter chunks without the merging step. Since most recordings in the AMI corpus have 4 speakers, which is close to $N = 3$, the baseline performance is relatively strong. The results in Table IX reveal that using shorter chunks leads to a higher MS+FA (20.03% vs. 18.92%) compared to the baseline. This indicates that short segments are more likely to miss active speakers, particularly in the presence of overlapping speech. On the other hand, shorter chunks yield a lower SC (3.18% vs. 4.32%), suggesting improved control in tracking speakers and avoiding redundant or incorrect assignments. Applying the merging step (i.e., the full proposed pipeline) with L_c of 15s results in the best overall performance. This

underscores the merging step as a crucial component that effectively combines the strengths of both long and short chunking strategies, while also highlighting its essential role within the overall inference framework.

Fig. 4 shows the performance comparison between the baseline and our proposed strategy with varying chunk sizes and merge sizes. The proposed pipeline consistently outperforms the baseline across various values of L_c, especially at larger durations. The degradation in the baseline's performance as chunk size increases can be attributed to an inappropriate choice of N. In contrast, the proposed method remains stable and achieves significantly lower DER without depending on the choice of N. These findings validate the effectiveness and reliability of our proposed pipeline, demonstrating strong performance while being less dependent on dataset-specific characteristics.

VI. Conclusion

In this work, we propose SUDx, a speaker-conditioned U-net based architecture that integrates hierarchical speaker representation to jointly address speaker diarization and extraction. Additionally, we introduce a novel inference pipeline, allowing the model to handle unknown number of speakers and reducing the need for large-scale pretraining data. The experimental results show that our model achieved state-of-the-art performance for speaker diarization while maintaining quality speech extraction. Evaluation on real meeting corpora further confirms the robustness of our approach and highlights the potential of our proposed strategy as a new promising pipeline for speaker diarization. Based on our experiments, we hypothesize that a powerful separator module can compensate for the extraction loss caused by the downsampling process. In future work, we plan to validate this hypothesis to further enhance both diarization and extraction performance.

References

[1] 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.

[2] 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.

[3] M. Rouvier, G. Dupuy, P. Gay, E. Khoury, T. Merlin, and S. Meignier, "An open-source state-of-the-art toolbox for broadcast news diarization," in *Interspeech*, 2013.

[4] S. H. Yella, A. Stolcke, and M. Slaney, "Artificial neural network features for speaker diarization," in *2014 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2014, pp. 402–406.

[5] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal *et al.*, "The ami meeting corpus: A pre-announcement," in *International workshop on machine learning for multimodal interaction*. Springer, 2005, pp. 28–39.

[6] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke *et al.*, "The icsi meeting corpus," in *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, vol. 1. IEEE, 2003, pp. I–I.

[7] S. Watanabe, M. Mandel, J. Barker, E. Vincent, A. Arora, X. Chang, S. Khudanpur, V. Manohar, D. Povey, D. Raj *et al.*, "Chime-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings," in *Proc. CHiME 2020*, 2020, pp. 1–7.

[8] 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.

[9] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *IEEE Transactions on audio, speech, and language processing*, vol. 20, no. 2, pp. 356–370, 2012.

[10] 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.

[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] 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.

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

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

[15] I. Medennikov, M. Korenevsky, T. Prisyach, Y. Khokhlov, M. Korenevskaya, I. Sorokin, T. Timofeeva, A. Mitrofanov, A. Andrusenko, I. Podluzhny *et al.*, "Target-speaker voice activity detection: A novel approach for multi-speaker diarization in a dinner party scenario," in *Proc. Interspeech 2020*, 2020, pp. 274–278.

[16] J. Kalda, R. Marxer, T. Alumäe, H. Bredin *et al.*, "Pixit: Joint training of speaker diarization and speech separation from real-world multi-speaker recordings," in *Proc. odyssey 2024*, 2024, pp. 115–122.

[17] S. Maiti, Y. Ueda, S. Watanabe, C. Zhang, M. Yu, S.-X. Zhang, and Y. Xu, "Eend-ss: Joint end-to-end neural speaker diarization and speech separation for flexible number of speakers," in *2022 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2023, pp. 480–487.

[18] C. Boeddeker, A. S. Subramanian, G. Wichern, R. Haeb-Umbach, and J. Le Roux, "Ts-sep: Joint diarization and separation conditioned on estimated speaker embeddings," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 1185–1197, 2024.

[19] J. Ao, M. S. Yıldırım, R. Tao, M. Ge, S. Wang, Y. Qian, and H. Li, "Used: Universal speaker extraction and diarization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.

[20] M. Ge, C. Xu, L. Wang, E. S. Chng, J. Dang, and H. Li, "Spex+: A complete time domain speaker extraction network," in *Proc. Interspeech 2020*, 2020, pp. 1406–1410.

[21] K. Kinoshita, M. Delcroix, and N. Tawara, "Integrating end-to-end neural and clustering-based diarization: Getting the best of both worlds," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 7198–7202.

[22] ——, "Advances in integration of end-to-end neural and clustering-based diarization for real conversational speech," in *Proc. Interspeech 2021*, 2021, pp. 3565–3569.

[23] M. He, D. Raj, Z. Huang, J. Du, Z. Chen, and S. Watanabe, "Target-speaker voice activity detection with improved i-vector estimation for unknown number of speaker," in *Proc. Interspeech 2021*, 2021, pp. 3555–3559.

[24] H. Bredin, "pyannote. audio 2.1 speaker diarization pipeline: principle, benchmark, and recipe," in *24th INTERSPEECH Conference (INTERSPEECH 2023)*. ISCA, 2023, pp. 1983–1987.

[25] H. Zhao, H. Chen, J. Yu, and Y. Wang, "Continuous target speech extraction: Enhancing personalized diarization and extraction on complex recordings," in *2024 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2024, pp. 1–8.

[26] 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.

[27] Y. Fu, L. Cheng, S. Lv, Y. Jv, Y. Kong, Z. Chen, Y. Hu, L. Xie, J. Wu, H. Bu *et al.*, "Aishell-4: An open source dataset for speech enhancement, separation, recognition and speaker diarization in conference scenario," in *Proc. Interspeech 2021*, 2021, pp. 3665–3669.

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

[29] G. Sell, D. Snyder, A. McCree, D. Garcia-Romero, J. Villalba, M. Maciejewski, V. Manohar, N. Dehak, D. Povey, S. Watanabe *et al.*, "Diarization is hard: Some experiences and lessons learned for the jhu team in the inaugural dihard challenge." in *Interspeech*, 2018, pp. 2808–2812.

[30] 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.

[31] M. Diez, L. Burget, S. Wang, J. Rohdin, and J. Cernockỳ, "Bayesian hmm based x-vector clustering for speaker diarization," in *Interspeech*, 2019.

[32] X. Xiao, N. Kanda, Z. Chen, T. Zhou, T. Yoshioka, S. Chen, Y. Zhao, G. Liu, Y. Wu, J. Wu *et al.*, "Microsoft speaker diarization system for the voxceleb speaker recognition challenge 2020," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5824–5828.

[33] A. Zhang, Q. Wang, Z. Zhu, J. Paisley, and C. Wang, "Fully supervised speaker diarization," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6301–6305.

[34] M. Maciejewski, D. Snyder, V. Manohar, N. Dehak, and S. Khudanpur, "Characterizing performance of speaker diarization systems on far-field speech using standard methods," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5244–5248.

[35] D. Raj, Z. Huang, and S. Khudanpur, "Multi-class spectral clustering with overlaps for speaker diarization," in *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 582–589.

[36] M. Kolbæk, D. Yu, Z.-H. Tan, and J. Jensen, "Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 10, pp. 1901–1913, 2017.

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

[38] L. Samarakoon, S. J. Broughton, M. Härkönen, and I. Fung, "Transformer attractors for robust and efficient end-to-end neural diarization," in *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2023, pp. 1–8.

[39] Z. Chen, B. Han, S. Wang, and Y. Qian, "Attention-based encoder-decoder end-to-end neural diarization with embedding enhancer," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 1636–1649, 2024.

[40] F. Landini, T. Stafylakis, L. Burget *et al.*, "Diaper: End-to-end neural diarization with perceiver-based attractors," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.

[41] M. Härkönen, S. J. Broughton, and L. Samarakoon, "Eend-m2f: Masked-attention mask transformers for speaker diarization," in *Proc. Interspeech 2024*, 2024, pp. 37–41.

[42] H. Bredin and A. Laurent, "End-to-end speaker segmentation for overlap-aware resegmentation," in *Interspeech 2021*, 2021.

[43] A. Plaquet and H. Bredin, "Powerset multi-class cross entropy loss for neural speaker diarization," in *Proc. Interspeech 2023*, 2023, pp. 3222–3226.

[44] D. Wang, X. Xiao, N. Kanda, T. Yoshioka, and J. Wu, "Target speaker voice activity detection with transformers and its integration with end-to-end neural diarization," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[45] C.-Y. Cheng, H.-S. Lee, Y. Tsao, and H.-M. Wang, "Multi-target extractor and detector for unknown-number speaker diarization," *IEEE Signal Processing Letters*, vol. 30, pp. 638–642, 2023.

[46] M.-K. He, J. Du, Q.-F. Liu, and C.-H. Lee, "Ansd-ma-mse: Adaptive neural speaker diarization using memory-aware multi-speaker embedding," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 1561–1573, 2023.

[47] G. Yang, M. He, S. Niu, R. Wang, Y. Yue, S. Qian, S. Wu, J. Du, and C.-H. Lee, "Neural speaker diarization using memory-aware multi-speaker embedding with sequence-to-sequence architecture," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 11 626–11 630.

[48] S. Wisdom, E. Tzinis, H. Erdogan, R. Weiss, K. Wilson, and J. Hershey, "Unsupervised sound separation using mixture invariant training," *Advances in neural information processing systems*, vol. 33, pp. 3846–3857, 2020.

[49] N. Saleem, T. S. Gunawan, S. Dhahbi, and S. Bourouis, "Time domain speech enhancement with cnn and time-attention transformer," *Digital Signal Processing*, vol. 147, p. 104408, 2024.

[50] S. He, H. Zhang, W. Rao, K. Zhang, Y. Ju, Y. Yang, and X. Zhang, "Hierarchical speaker representation for target speaker extraction," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 10 361–10 365.

[51] S. He, W. Xue, Y. Yang, H. Zhang, J. Pan, and X. Zhang, "Enhancing target speaker extraction with hierarchical speaker representation learning," *Neural Networks*, vol. 188, p. 107388, 2025.

[52] Y. Ju, J. Chen, S. Zhang, S. He, W. Rao, W. Zhu, Y. Wang, T. Yu, and S. Shang, "Tea-pse 3.0: Tencent-ethereal-audio-lab personalized speech enhancement system for icassp 2023 dns-challenge," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–2.

[53] C. Xu, W. Rao, E. S. Chng, and H. Li, "Spex: Multi-scale time domain speaker extraction network," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 28, pp. 1370–1384, 2020.

[54] W. Liu and C. Xie, "Gated convolutional fusion for time-domain target speaker extraction network." in *INTERSPEECH*, 2022, pp. 5368–5372.

[55] Y. Luo and N. Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[56] E. Tzinis, Z. Wang, and P. Smaragdis, "Sudo rm-rf: Efficient networks for universal audio source separation," in *2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2020, pp. 1–6.

[57] K. Li, R. Yang, and X. Hu, "An efficient encoder-decoder architecture with top-down attention for speech separation," in *The Eleventh International Conference on Learning Representations*, 2022.

[58] Z. Pan, M. Ge, and H. Li, "Usev: Universal speaker extraction with visual cue," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 3032–3045, 2022.

[59] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr–half-baked or well done?" in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 626–630.

[60] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *Interspeech 2020*, 2020.

[61] J. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *Interspeech 2018*, 2018.

[62] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, L. Lugosch, C. Subakan, N. Dawalatabad, A. Heba, J. Zhong *et al.*, "Speechbrain: A general-purpose speech toolkit," *arXiv preprint arXiv:2106.04624*, 2021.

[63] F. Hao, X. Li, and C. Zheng, "X-tf-gridnet: A time–frequency domain target speaker extraction network with adaptive speaker embedding fusion," *Information Fusion*, vol. 112, p. 102550, 2024.

[64] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, vol. 33, pp. 12 449–12 460, 2020.

[65] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 29, pp. 3451–3460, 2021.

[66] S.-w. Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin *et al.*, "Superb: Speech processing universal performance benchmark," *Interspeech 2021*, 2021.

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

[68] S. Horiguchi, Y. Fujita, S. Watanabe, Y. Xue, and P. Garcia, "Encoder-decoder based attractors for end-to-end neural diarization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 1493–1507, 2022.

[69] N. Ryant, P. Singh, V. Krishnamohan, R. Varma, K. Church, C. Cieri, J. Du, S. Ganapathy, and M. Liberman, "The third dihard diarization challenge," in *Proc. Interspeech 2021*, 2021, pp. 3570–3574.

Ngoc Thuan Tran received the B.S. degree in control engineering and

automation from the Hanoi University of Science and Technology (HUST), Hanoi, Vietnam, where he is currently working toward the M.S. degree in control engineering and automation with the School of Electrical and Electronic Engineering. His research interests include signal processing, speaker diarization, and target speaker extraction.

Ngoc Chau Hoang received the B.S. and M.S. degree in control engineering and automation from the Hanoi University of Science and Technology (HUST), Hanoi, Vietnam, in 2022 and 2024, respectively. He is currently an AI engineer with Viettel AI, Viettel Group, Vietnam. His research interests include speech processing, source separation, and automatic speech recognition.

Quoc Cuong Nguyen received the engineer and M.S. degrees in electrical engineering from the Hanoi University of Science and Technology (HUST), Hanoi, Vietnam, in 1996 and 1998, respectively, and the Ph.D. degree in signal-image-speech-telecoms from the Institut National Polytechnique de Grenoble (INP Grenoble), France, in 2002. He is currently a Lecturer/Researcher with the School of Electrical and Electronic Engineering, HUST. His research interests include signal processing, speech recognition, embedded systems, and RF communication.