

Listen to Extract: Onset-Prompted Target Speaker Extraction

Pengjie Shen, Kangrui Chen, Shulin He, Pengru Chen, Shuqi Yuan, He Kong[✉], Xueliang Zhang[✉], and Zhong-Qiu Wang[✉]

Abstract—We propose *listen to extract* (LExt), a highly-effective while extremely-simple algorithm for monaural target speaker extraction (TSE). Given an enrollment utterance of a target speaker, LExt aims at extracting the target speaker from the speaker’s mixed speech with other speakers. For each mixture, LExt concatenates an enrollment utterance of the target speaker to the mixture signal at the waveform level, and trains deep neural networks (DNN) to extract the target speech based on the concatenated mixture signal. The rationale is that, this way, an artificial speech onset is created for the target speaker and it could prompt the DNN (a) which speaker is the target to extract; and (b) spectral-temporal patterns of the target speaker that could help extraction. This simple approach produces strong TSE performance on multiple public TSE datasets including WSJ0-2mix, WHAM! and WHAMR!.

Index Terms—Target speaker extraction, onset-prompted speech separation.

I. INTRODUCTION

IN MANY artificial intelligence and machine learning applications, the sensors inevitably record a mixture of target and non-target signals. The recorded non-target signals often pose tremendous difficulties in the perception and understanding of the target signals. In this case, how to extract and enhance the target signal(s) is an important research problem to study. One example, in audio signal processing, is speaker separation (*a.k.a.*, the cocktail party problem) [1], [2], [3], where, in a noisy-reverberant room with multiple speakers talking concurrently, the task is to extract each of the target speakers of interest based on the mixtures recorded in the room. In the past decade, dramatic progress has been made in speaker separation [3], [4], [5], thanks to the rapid development of deep learning. Some studies

Received 8 May 2025; revised 7 October 2025; accepted 3 November 2025. Date of publication 17 November 2025; date of current version 24 November 2025. The associate editor coordinating the review of this article and approving it for publication was Dr. Jonas Borgstrom. (*Corresponding author:* Zhong-Qiu Wang.)

Pengjie Shen is with the Department of Computer Science, Inner Mongolia University, Hohhot 010021, China (e-mail: shenpengjie@mail imu.edu.cn).

Xueliang Zhang is with the Department of Computer Science, Inner Mongolia University, Hohhot 010021, China (e-mail: cszxl@imu.edu.cn).

Kangrui Chen, Shulin He, Pengru Chen, Shuqi Yuan, and Zhong-Qiu Wang are with the Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China (e-mail: chenkr2021@mail.sustech.edu.cn; hesl@sustech.edu.cn; wang.zhongqiu41@gmail.com).

He Kong is with the School of Automation and Intelligent Manufacturing, Southern University of Science and Technology, Shenzhen 518055, China (e-mail: kongh@sustech.edu.cn).

Digital Object Identifier 10.1109/TASLPRO.2025.3633088

train deep neural networks (DNN) to separate all the speakers in the mixture via permutation invariant training (PIT) [4], [5], where the input to the DNN is a multi-speaker mixture and the DNN is trained, in a supervised way, to estimate all the speakers in the mixture.

Differently, some other studies only aim at target speaker extraction (TSE), which targets at extracting a particular speaker of interest by providing to the DNN some auxiliary cues that can indicate which speaker is the targeted one to extract [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19]. TSE is an important problem to study, since in many application scenarios one may only care about a particular target speaker of interest. One approach to realize TSE is to first separate all the speakers, and then identify the speaker estimate corresponding to the target speaker [11]. However, as is discussed in [11], this approach often produces suboptimal performance compared to TSE for the following reasons: (a) separating all the speakers is a difficult task, especially when the number of mixed speakers is large; (b) the separated speakers suffer from permutation ambiguity (i.e., an arbitrary assignment of speaker estimates to true target speakers) and subsequent effort is needed to identify the speaker estimate corresponding to the target speaker; and (c) the number of speakers is typically unknown and can vary a lot in different mixtures, creating difficulties for separating all the speakers, as speaker counting may not be perfect.

Many auxiliary cues can be utilized for TSE, such as an enrollment utterance or a speaker embedding [13], [19], visual lip movements [20], [21], bone-conduction signals [22], speaker-activity timestamps [23], direction information [24] and brain signals [25], [26] of the target speaker. Among various auxiliary cues, enrollment utterances are typically easy to obtain, as users can easily register their voice to the TSE system by, for example, speaking something to their cell phones. With this observation, in this paper we investigate TSE given an enrollment utterance of the target speaker. Previous algorithms in this direction [10], [11], [14], [17] usually extract a fixed-length speaker embedding from the enrollment utterance, and use it to condition a speaker extraction network at one or multiple layers to extract the target speaker. In the past decade, many conditioning mechanisms have been proposed, such as performing concatenation [12], summation [14], [27], multiplication [6], cross-attention [28], [29], and feature-wise linear modulation (FiLM) [30], [31] operations between the fixed-length speaker embedding and the internal tensors produced by the speaker extraction network. A popular way to extract the fixed-length speaker embedding is by

using pre-trained speaker recognition models [13] such as the x-vector [32] and ECAPA-TDNN [33] models. Another way is to train the speaker extraction network with a speaker embedding network through multi-task learning, where the speaker embedding network can be initialized by using a pre-trained speaker recognition model and a speaker classification loss can be attached to maintain the speaker discrimination capabilities of the pre-trained speaker recognition model [6], [19]. Although these two approaches have been very popular in recent years in TSE research, they have two issues [10], [11], [17]. First, the fixed-length speaker embedding may mismatch the tensors produced inside the speaker extraction network, leading to difficulties in achieving highly-efficient speaker conditioning. Second, the fixed-length speaker embedding is designed to be capable of discriminating different speakers for speaker recognition, but it may not be optimal for the final TSE task.

To deal with the above issues, a stream of research attempts to not rely on fixed-length speaker embeddings, and instead designs a way to leverage variable-length speaker embeddings so that the temporal patterns and local dynamics of the target speaker in the enrollment utterance can be exploited for TSE. Xiao et al. [34] first embed the mixture and the enrollment utterance, obtaining an embedding sequence for the mixture and another for the enrollment utterance, and then a cross-attention operation, which uses each embedding in the mixture embedding sequence as query and the embeddings in the enrollment sequence as key and value, is applied to obtain a new embedding sequence, which, by design, has the same length as the mixture embedding sequence. Next, the new embedding sequence is combined with the mixture embedding sequence for subsequent DNN modules to extract the target speaker. Notice that the new embedding sequence can be considered as variable-length speaker embeddings and could be more matched with the mixture embeddings for speaker conditioning, compared with fixed-length speaker embeddings.

This research has motivated subsequent studies such as CIENet [35], [36] and USEF-TSE [37]. CIENet [35], [36] uses raw short-time Fourier transform (STFT) coefficients as the query, key and value tensors for the cross-attention operation proposed in [34], and in addition employs a stronger speaker extraction network. USEF-TSE [37], an improved version of SEF-Net [38], improves Xiao et al.'s approach by employing an advanced cross-attention mechanism proposed in TF-GridNet [39] to obtain variable-length speaker embeddings for speaker conditioning and by using an advanced DNN architecture (i.e., TF-GridNet) for speaker extraction. USEF-TSE reports strong performance on multiple TSE benchmarks.¹

In this context, we propose another TSE algorithm free of fixed-length speaker embeddings named *listen to extract* (LExt). Compared with existing studies, it is much simpler while obtaining better TSE performance. It concatenates an enrollment utterance of the target speaker to the input mixture (e.g., by

¹USEF-TSE [37] and its preliminary version SEF-Net [38] argue that they are *speaker-embedding-free*, since they do not rely on conventional speaker embeddings, which are usually fixed-length. However, one may consider that a variable-length speaker embedding is extracted in their system. As a result, this paper avoids using the term *speaker-embedding-free*.

prepending the enrollment utterance to the mixture at the waveform level), and trains a DNN to extract the target speaker based on the concatenated mixture. The idea is to have the DNN first listen to an example utterance of the target speaker, which may, later on, prompt the DNN to extract another utterance of the target speaker from the observed mixture. We highlight that LExt does not require customized DNN architectures to realize such a speaker conditioning. Many existing DNN architectures in speech separation (e.g., TF-GridNet [39], SepFormer [40] and TF-LocoFormer [41]) could be straightforwardly employed, if the DNN (a) has a sufficiently-large receptive field (e.g., realized by self-attention) to see the concatenated enrollment utterance while extracting the target speaker from the input mixture; or (b) has a memory mechanism such as long short-term memory (LSTM) that can memorize what the DNN has listened to in the beginning of the prepended mixture. In our experiments on multiple public datasets designed for TSE (including WSJ0-2mix [5], WHAM! [42] and WHAMR! [43]), LExt achieves strong TSE performance, despite being very simple. A sound demo is provided in the link below.²

Why would such a simple approach work? There could be two reasons:

- The first reason is related to *onset-based speaker separation* [44], which aims at separating all the speakers (but is not designed for TSE). Instead of using PIT to address the permutation ambiguity problem, it orders all the speakers according to their speech onset (e.g., from the earliest to latest) and trains DNNs to predict all the speakers according to this order. In [44], the authors observe strong performance of this onset-based approach for two-speaker separation. In an earlier study conducted by the same research group, the authors propose attentive training [45], where a DNN is trained to estimate the speaker signal with the earliest speech onset, and the training has been shown very successful. In LExt, the prepended enrollment utterance manually creates, for the target speaker, a speech onset earlier than all the other speakers. This earliest onset could make the extraction of the target speaker possible.
- The second reason could be that, in LExt, at every DNN layer, the hidden representations extracted from the enrollment utterance are *homogeneous* with those extracted from the mixture, simply because they are extracted by exactly the same DNN modules. This way, the speaker conditioning in LExt could be more effective, compared with just conditioning some layers of the speaker extraction DNN by speaker embeddings, which are extracted by other DNN modules (e.g., a DNN trained for speaker recognition) and hence may be much less homogeneous with the hidden representations produced by the speaker extraction DNN. In addition, in LExt, implicit speaker conditioning exists at *every layer* of the DNN, as we use exactly the same DNN to process every frame of the concatenated mixture. This could be a much more fine-grained speaker conditioning mechanism than speaker-embedding-based

²See https://zqwang7.github.io/demos/LExt_demo/index.html.

methods, which usually only perform speaker-embedding-based conditioning at a shallow layer [10], [11], [17] or at a selected subset of intermediate layers [31], [38], [46], [47].

The remainder of this paper is organized as follows. Section II discusses related works, III proposes LExt, IV presents experimental setup, V reports evaluation results, VI discusses limitations, and VII draws conclusions.

II. RELATED WORKS

LExt is related to existing studies mainly in the following aspects.

A. TSE Methods Not Using Fixed-Length Speaker Embeddings

The approach proposed by Xiao et al. [34], SEF-Net [38], CIENet [35], [36], and SEF-Net's improved version USEF-TSE [37] are representative TSE algorithms that do not rely on fixed-length speaker embeddings. We have described them in the introduction section. Compared with them, LExt is a much simpler algorithm while obtaining better TSE performance. This is possibly due to the implicit speaker conditioning mechanism in LExt, which can happen at every layer of the DNN. In comparison, in, e.g., USEF-TSE [37], the cross-attention layer, which is the key for speaker conditioning, is placed at a shallow layer, operating on the embeddings obtained by applying a linear encoder to the spectrograms of the mixture and enrollment utterance. Such a mechanism may not be very effective at speaker conditioning.

B. Leveraging Estimated Onset for TSE

In [48], the speech onset of the target speaker is estimated based on the input mixture and enrollment utterance, and the estimated onset is encoded as a binary vector (which has the same length as the mixture) to improve TSE systems that are based on fixed-length speaker embeddings. Very differently, LExt creates an artificial onset to realize TSE.

C. Onset-Prompted Overlapped Speech Processing

There are existing algorithms based on onset-prompting for multi-speaker overlapped speech processing [44], [45], [49]. One example is in speaker separation, where the task is to separate the mixture of multiple speakers to individual speakers and the DNN model needs to resolve permutation ambiguity (i.e., figuring out how to align estimated speakers with ground-truth target speakers to realize successful training). Differently from using a popular solution named PIT [4], [5], which first aligns estimates with labels before loss computation, the authors in [44], [45] propose to order DNN estimates according to the onset information of different speakers. Strong speaker separation performance is observed. Another example [49] is in multi-speaker automatic speech recognition, where the task is to recognize the speech of each speaker in multi-speaker mixtures. The authors propose a solution named serialized output training [49], which orders DNN-estimated token sequences according to the onset cues of the mixed speakers. The rationale of the above studies is

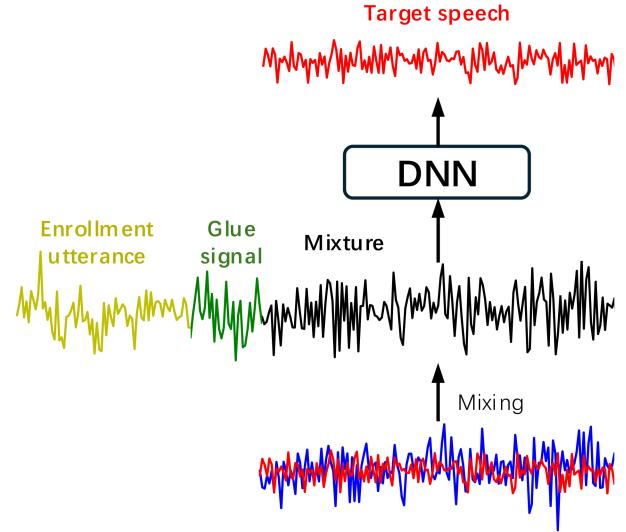


Fig. 1. Illustration of LExt for TSE. In this example, an enrollment utterance and a glue signal are prepended to the mixture for the DNN to extract the target speaker. Best viewed in color. The mixture signal (shown in black) consists of the target speech (in red) and non-target signals (in blue).

that different speakers usually start talking at different time, and such differences in onset can be leveraged by carefully-designed DNN modules to resolve permutation ambiguity.

Different from existing onset-based overlapped speech processing methods [44], [45], [49], where the goal is to separate or recognize all the mixed speakers, LExt deals with TSE, a task where there is a single target speaker of interest to extract. In addition, LExt proposes to introduce an *artificial* onset for the target speaker by, e.g., prepending an enrollment utterance to the mixture. These differences set LExt apart from existing onset-prompted algorithms in overlapped speech processing.

III. LEXT

Given a noisy-reverberant environment with C speakers, the physical model of the mixture signal captured by a far-field microphone can be formulated, in the time domain, as

$$y = s + v \in \mathbb{R}^N, \quad (1)$$

where N denotes the number of time-domain samples (i.e., signal length), and y , s , and v respectively represent the captured mixture, target speech by a desired target speaker we aim to extract, and non-target signals. We use v to absorb all the signals that are not considered as target speech, including the reverberation of the target speaker, environmental noises, and signals produced by the other speakers. In the task of TSE, an E -sample-long enrollment utterance $e \in \mathbb{R}^E$ is assumed available. It is an utterance different from the target speech s , but is uttered by the same speaker as s . This way, it can indicate who the target speaker is and hence can be utilized to help extract the target speaker from the mixture. The rest of this section describes our proposed LExt algorithm and its design choices.

A. Listen to Extract (LExt)

The proposed LExt approach is illustrated in Fig. 1. This embodiment of LExt prepends an enrollment utterance e to the mixture y . They are concatenated by using a G -sample-long glue signal $g \in \mathbb{R}^G$. The augmented mixture signal is fed to a DNN, which is trained to extract the target speaker indicated by the enrollment utterance.

To train the DNN, for each clean and mixture signal pair $\langle s, y \rangle$ ³ used for training, we first augment each of the two signals to create a new pair $\langle \tilde{s}, \tilde{y} \rangle$, where $\tilde{y} = [e; g; y] \in \mathbb{R}^{E+G+N}$ prepends an enrollment utterance e and a glue signal g to the mixture y , and $\tilde{s} = [e; g; s] \in \mathbb{R}^{E+G+N}$ prepends the same enrollment utterance e and the same glue signal g to the target speech s . The new pair is then utilized to train the DNN via supervised learning.

B. Length and Values of Glue Signal

In LExt, a glue signal g , described in the previous subsection, is utilized to prompt the DNN the time ranges of the enrollment utterance and the mixture. Since a longer glue signal increases the length of the augmented mixture and would require more computation, we set it to only 32 ms long. On the other hand, for simplicity, we set it to an all-zero signal.

C. Loss Functions

The loss function is defined on the estimated target speech after discarding the predictions in the time range of the concatenated enrollment and glue signals (i.e., on $\hat{s} = \hat{s}[E + G :]$, where $[E + G :]$ means removing the first $E + G$ samples). Any off-the-shelves loss functions in TSE can be used. Our study just chooses the scale-invariant signal-to-distortion ratio (SI-SDR) loss [50], which is widely-adopted in TSE research.

D. DNN Architectures

LExt can work with many modern DNN architectures in speech separation and enhancement, as long as the DNN can have a mechanism (such as self-attention) that can see the prepended enrollment utterance (at the very beginning of the augmented mixture), while extracting the target speaker.

In our experiments, we investigate two state-of-the-art separation models for LExt, TF-GridNet [39] and TF-LocoFormer [41]. Their configurations are described later in Section IV-B. Note that, to enable the speaker-conditioning mechanism in LExt, we do not need to modify those DNN architectures, and instead we just need to change the input and output of those DNN architectures and the DNN could learn to leverage the artificial onset to extract the target speaker. This property distinguishes LExt from existing TSE methods [10], [11], [17], which need to modify the DNN architectures internally and add some internal DNN modules to enable speaker conditioning.

Both networks are trained via complex spectral mapping [51], [52], [53] to predict the real and imaginary (RI)

³The notation, $\langle \cdot, \cdot \rangle$, denotes a signal pair consisting of a mixture input signal and its corresponding target signal.

components of the target speech based on the RI components of the input signal.

E. Gain Normalization of Enrollment Utterance and Mixture

We find it beneficial to balance the gain of the mixture y and the enrollment utterance e before concatenating them.

During training, we normalize the sample variance of each signal to 1.0 before concatenation. In detail, given the mixture signal y , target speech s and enrollment utterance e , we normalize each of them in the following way, before creating an augmented pair $\langle \tilde{y}, \tilde{s} \rangle$ for training:

$$s := s / \sigma(y), \quad (2)$$

$$y := y / \sigma(y), \quad (3)$$

$$e := e / \sigma(e), \quad (4)$$

where $\sigma(\cdot)$ computes the standard deviation of the signal at the sample level.

At run time, we do the same normalization to y and e before concatenating them for inference. After obtaining target estimate \hat{s} , we multiply the samples in the time range of the enrollment utterance with $\sigma(e)$, and multiply the samples in the mixture range with $\sigma(y)$ to reverse the normalization.

F. Only Prepend, or Not Only Prepend but Also Append?

So far, we assume that the enrollment utterance is prepended to the mixture signal. Alternatively, we can prepend half of the enrollment utterance to the mixture while append the rest. This could be helpful for DNN architectures such as TF-GridNet [39] but not for architectures such as TF-LocoFormer [41], since TF-GridNet relies heavily on bi-directional LSTM (BLSTM) modules, which may have a limited receptive field [54], while TF-LocoFormer has many self-attention blocks, which lead to a large receptive field.

G. Removing Silences in Enrollment Utterance

For each enrollment utterance, we use a pre-trained speech activity detector,⁴ which is based on the Kaldi toolkit, to remove the silences in the enrollment utterance, as they are not informative about the speaker characteristics of the target speaker. In addition, removing them can reduce the computation of LExt, as the length of the enrollment utterance is reduced. For the identified silence segments, we simply discard them and splice together non-silent segments.

IV. EXPERIMENTAL SETUP

We validate the effectiveness of LExt at TSE based on public datasets, including WSJ0-2mix, WHAM! and WHAMR!, all of which have been widely-adopted in previous TSE studies. This section describes the datasets, system configurations of LExt, evaluation metrics, and baseline systems.

⁴<https://kaldi-asr.org/models/m4>

A. Datasets

The WSJ0-2mix [5], WHAM! [42] and WHAMR! [43] are all originally designed for talker-independent multi-speaker separation. They are later modified for TSE, by specifying each one of the speakers in each mixture in turn as the target speaker and specifying an enrollment utterance for the target speaker. For all the three datasets, we use the same enrollment utterances⁵ as earlier studies for evaluation. The average length of the enrollment utterances is ~ 7.3 seconds. For all the datasets, we use the *min* and 8 kHz version.

WSJ0-2mix [5] is so far the most popular dataset to evaluate monaural talker-independent speaker separation algorithms in anechoic conditions. It consists of 20,000 (~ 30.4 h), 5,000 (~ 7.7 h) and 3,000 (~ 4.8 h) two-speaker mixtures in its training, validation and test sets, respectively. The clean source signals are sampled from the WSJ0 corpus. The speakers for training and validation do not overlap with the speakers for testing. The two utterances in each mixture are fully-overlapped, and their relative energy level is uniformly sampled from the range $[-5, 5]$ dB.

WHAM! [42], building on the WSJ0-2mix dataset, introduces environmental noises, extending the two-speaker separation task in WSJ0-2mix to noisy conditions. The noise dataset used for mixing consists of 80 hours of real-recorded signals in urban areas such as coffee shops, restaurants, bars, parks and office buildings. For each two-speaker mixture in WSJ0-2mix, a randomly-selected noise signal is added such that the signal-to-noise ratio (SNR) of the louder speaker and the noise is equal to a value randomly sampled from the range $[-6, 3]$ dB. WHAM! consists of 20,000 (~ 30.4 h), 5,000 (~ 7.7 h) and 3,000 (~ 4.8 h) noisy two-speaker mixtures in its training, validation and test sets, respectively.

WHAMR! [43] is used to validate our algorithms in noisy-reverberant conditions. It is based on the two-speaker mixtures in WSJ0-2mix [5] but reverberates each clean speaker source and adds non-stationary noises. In each mixture, the reverberation time (T60) is randomly sampled from the range $[0.2, 1.0]$ s, the SNR between the louder speaker and noise is drawn from $[-6, 3]$ dB, relative energy level between the two speakers from $[-5, 5]$ dB, and speaker-to-array distance from $[0.66, 2.0]$ m. There are 20,000 (~ 30.4 h), 5,000 (~ 7.7 h) and 3,000 (~ 4.8 h) binaural mixtures for training, validation and testing, respectively. The direct-path signal of the target speaker at the first microphone is considered as the target signal for training and as the reference signal for metric computation. We only use the first microphone for training and evaluation. In this TSE task, the model needs to jointly reduce room reverberation, environmental noises and competing speech.

B. DNN Configurations

We consider two DNN architectures for LExt, including TF-GridNet [39] and TF-LocoFormer [41], both of which operate in the time-frequency (T-F) domain and are representative speech

TABLE I
LEXT PERFORMANCE ON WSJ0-2MIX WHEN USED WITH VARIOUS ENROLLMENT UTTERANCE LENGTH

Row	System	DNN arch.	Prepend length (s)	SI-SDRi (dB)†
1Z	LExt	TFGridNetV1	0.10	18.7
1A	LExt	TFGridNetV1	0.25	20.8
1B	LExt	TFGridNetV1	0.50	22.0
1C	LExt	TFGridNetV1	1.00	22.3
1D	LExt	TFGridNetV1	2.00	22.8
1E	LExt	TFGridNetV1	4.00	23.0
1F	LExt	TFGridNetV1	6.00	22.9
2A	LExt	TFLocoFormerV1	0.50	18.2
2B	LExt	TFLocoFormerV1	1.00	20.2
2C	LExt	TFLocoFormerV1	2.00	20.3
2D	LExt	TFLocoFormerV1	4.00	19.2

separation models reporting strong separation performance in recent supervised speech separation benchmarks. TF-LocoFormer is a dual-path DNN architecture like TF-GridNet but replaces BLSTMs with transformer blocks. The transformer blocks can easily see the concatenated enrollment utterance via its self-attention mechanism.

For each architecture, we investigate two configurations, denoted as *V1* and *V2*. The *V1* version uses less computation for faster experimentation. Following the symbols defined in Table I of the TF-GridNet paper [39], for **TFGridNetV1** we set the hyper-parameters to $D = 128, B = 4, I = 1, J = 1, H = 200, L = 4$ and $E = 16$, and, for **TFGridNetV2**, we set them to $D = 128, B = 6, I = 1, J = 1, H = 256, L = 4$ and $E = 16$. Following the TF-LocoFormer paper [41], we employ its small version, denoted as **TFLocoFormerV1**, and its medium version, denoted as **TFLocoFormerV2**. Please do not confuse the symbols used by TF-GridNet and TF-LocoFormer with the ones defined in this paper.

C. Miscellaneous Configurations of LExt

The model is optimized by using the Adam optimizer for TF-GridNet, while by using the AdamW optimizer for TF-LocoFormer, following the suggestions by the original authors. For STFT, the square root of Hann window is used as the analysis window, and for TF-GridNet, the window size and hop size are respectively set to 16 and 8 ms, while for TF-LocoFormer, they are respectively set to 32 and 16 ms to speed up training.

For each training mixture, at each epoch, we randomly choose one of the mixed speakers as the target speaker, and select another utterance of the target speaker as the enrollment utterance. We train LExt by using fixed-length signal segments sampled from the original mixtures and enrollment utterances. The length of the sampled mixture segments is 4.0 seconds, while the length of the sampled enrollment segments is a tunable hyper-parameter. During training, the length of the enrollment segments and that of the mixture segments are both fixed. At inference time, the length of the enrollment utterance is fixed and configured the same as the length used during training, while the mixture is in its full length. Our motivations for designing the enrollment as fixed-length are two-fold. First, since LExt concatenates the enrollment directly with the mixture as the network input, using the original, full-length utterance as the

⁵See <https://github.com/ZBang/USEF-TSE/tree/main/data/test/>.

enrollment would make the overall input sequence to the network excessively long. This not only increases computation but also risks the model forgetting or under-utilizing the enrollment information, especially in scenarios where the original enrollment utterance itself is long. Second, we found that adopting a fixed-length enrollment improves training stability by reducing the variability introduced by using enrollment segments of different lengths. This ensures more consistent training conditions and mitigates uncertainties caused by highly-variable input lengths.

D. Evaluation Metrics

Our evaluation metrics follow existing TSE studies. We use SI-SDR [50] as the main evaluation metric. We additionally employ BSS-Eval SDR [55] and narrow-band perceptual evaluation of speech quality (PESQ) [56]. All of them are widely-adopted in TSE research. For SI-SDR and SDR, we report scores in SI-SDR improvement (SI-SDR_i) and SDR improvement (SDR_i) over unprocessed mixtures. The scores are averaged over the mixtures in the test set.

E. Baseline Systems

There are two main categories of baseline systems, one based on variable-length speaker embeddings and another based on fixed-length speaker embeddings.

We compare LExt with USEF-TSE [37], which leverages variable-length speaker embeddings for speaker conditioning and has been described in the introduction section. It builds upon TF-GridNet [39] for TSE. We emphasize that TFGridNetV2 used in this paper is the same as the TF-GridNet architecture used in the original USEF-TSE paper, except for the following two differences. First, USEF-TSE has an additional speaker conditioning module, based on cross-attention, to enable TF-GridNet to perform TSE. Second, in USEF-TSE, the embedding dimension of each T-F unit is set to 256 (as it stacks the embedding of the mixture and the variable-length speaker embedding, both of which are 128-dimensional at each T-F unit), while, in LExt, it is 128.

We also compare LExt with TSE systems based on fixed-length speaker embeddings. We employ such a system suggested by the WeSep toolkit [14], where a speaker embedding network (i.e., ECAPA-TDNN [33]) is used to extract a fixed-length speaker embedding from the enrollment utterance and the speaker embedding is used to condition a TF-GridNet based speaker extraction network at multiple layers. Following [14], we remove the speaker classification loss of ECAPA-TDNN, and the speaker embedding network and the speaker extraction network are jointly trained to optimize the SI-SDR loss. See Section 3 of the WeSep paper [14] for more details.

On the other hand, since LExt is evaluated on public datasets, its evaluation results can be directly compared with many existing ones to show its effectiveness.

V. EVALUATION RESULTS

We first present ablation results based on the WSJ0-2mix dataset to validate various design choices of LExt, and then report the performance of LExt on multiple public benchmarks.

TABLE II
LEXT PERFORMANCE ON WSJ0-2MIX WHEN PREPENDING VS. BOTH PREPENDING AND APPENDING ENROLLMENT UTTERANCE

System	DNN arch.	Prepend length (s)	Append length (s)	SI-SDR _i (dB) \uparrow	
				Validation	Test
LExt	TFGridNetV1	4.0	-	22.2	23.0
LExt	TFGridNetV1	2.0	2.0	22.1	22.9
LExt	TFGridNetV1	-	4.0	22.2	22.8
LExt	TFLocoFormerV1	4.0	-	19.7	19.2
LExt	TFLocoFormerV1	2.0	2.0	19.0	18.0
LExt	TFLocoFormerV1	-	4.0	19.2	17.8
LExt	TFLocoFormerV1 (NoPE)	4.0	-	19.8	18.3
LExt	TFLocoFormerV1 (NoPE)	2.0	2.0	19.7	18.6
LExt	TFLocoFormerV1 (NoPE)	-	4.0	19.9	18.8

TABLE III
LEXT PERFORMANCE ON WSJ0-2MIX WHEN USING V1 VS. V2 MODELS

System	DNN arch.	Prepend length (s)	SI-SDR _i (dB) \uparrow	SDR _i (dB) \uparrow	PESQ \uparrow
Mixture	-	-	0.0	0.0	1.68
LExt	TFGridNetV1	4.0	23.0	23.2	4.05
LExt	TFGridNetV2	4.0	24.1	24.3	4.10
LExt	TFLocoFormerV1	4.0	19.2	19.5	3.78
LExt	TFLocoFormerV2	4.0	21.9	22.1	3.98

A. Effects of Length of Enrollment Utterance

In Table I, we compare the results of using enrollment utterances with various durations for speaker conditioning. Each result is obtained by fixing the lengths of all the enrollment utterances to E seconds during training and testing, where $E \in \{0.1, 0.25, 0.5, 1.0, 2.0, 4.0, 6.0\}$ seconds.

After the silence segments are removed using speech activity detection (SAD), some enrollment utterances are shorter than E seconds. In this case, we pad zeros to the left, rather than to the right, of the enrollment utterance before prepending it to the mixture, considering that, this way, the active speech in the enrollment utterance can be closer to the mixture and this padding strategy could benefit models such as TF-GridNet which relies heavily on BLSTM-based sequential modules. After removing the silence segments using SAD, for the enrollment utterances still longer than E seconds, we randomly sample an E -second segment in each training epoch for model training, while we always use the first E -second segment for evaluation.

From the results in Table I, we observe that LExt does not require very long enrollment utterance. For example, using TFGridNetV1 and only 0.5-second-long enrollment speech can yield 22.0 dB SI-SDR_i. This result is already better than many existing TSE systems (shown later in Table IV), which leverage enrollment utterances in their full length (on average ~ 7.3 seconds) for speaker conditioning. For TFGridNetV1, the SI-SDR_i result becomes better when the length of the enrollment utterance increases from 0.1 to 4.0 seconds. This is reasonable as longer enrollment utterance can offer more cues about the target speaker. When the length of the enrollment utterance is further increased from 4.0 to 6.0 seconds, the performance is not improved and appears saturated. The lack of monotonic improvement with longer enrollments is possibly due to the statistics of the enrollment lengths of the WSJ0-2mix dataset. After SAD processing, the average enrollment length is about 4.53 seconds in the training set and 5.57 seconds in the test

TABLE IV
SI-SDR_I (dB)↑, SDR_I (dB)↑ AND PESQ↑ COMPARISON OF LEXT WITH EXISTING ALGORITHMS BASED ON WSJ0-2MIX, WHAM! AND WHAMR!

Row	System	Task	WSJ0-2mix			WHAM!			WHAMR!		
			SI-SDR _I ↑	SDR _I ↑	PESQ↑	SI-SDR _I ↑	SDR _I ↑	PESQ↑	SI-SDR _I ↑	SDR _I ↑	PESQ↑
0	Mixture	-	0.0	0.0	1.68	0.0	0.0	1.43	0.0	0.0	1.42
1A	Conv-TasNet [57]		15.3	15.6	-	12.7	-	-	8.3	-	-
1B	DPRNN [58]		18.8	19.0	-	13.9	-	-	10.3	-	-
1C	WaveSplit [59]		21.0	21.2	-	15.4	15.8	-	12.0	11.1	-
1D	WaveSplit+DM [59]		22.2	22.3	-	16.0	16.5	-	13.2	12.2	-
1E	SepFormer [40]		20.4	20.5	-	14.7	15.1	-	11.4	10.4	-
1F	SepFormer+DM [40]		22.3	22.4	-	16.4	16.7	-	14.0	13.0	-
1G	QDPN [60]		22.1	-	-	-	-	-	13.1	-	-
1H	DPTNet [61]	SS	20.2	20.6	-	-	-	-	-	-	-
1I	MossFormer(L)+DM [62]		22.8	-	-	17.3	-	-	16.3	-	-
1J	MossFormer2(L)+DM [63]		24.1	-	-	18.1	-	-	17.0	-	-
1K	TD-Conformer-XL [64]		20.4	-	-	-	-	-	13.1	-	-
1L	TF-GridNet [39]		23.5	23.6	-	-	-	-	17.1	15.6	2.69
1M	TF-CrossNet [65]		23.2	23.4	-	-	-	-	17.9	16.4	2.91
2A	SpEx+ [66]		16.9	17.2	3.43	13.1	13.6	-	10.9	10.0	-
2B	DPRNN-Spe-IRA [67]		17.3	17.6	3.43	14.2	14.6	2.57	-	-	-
2C	SpEx++ [68]		17.9	18.3	3.52	14.0	14.4	-	11.4	10.4	-
2D	VEVEN [69]		19.0	19.2	-	-	-	-	-	-	-
2E	X-TF-GridNet [31]		20.7	21.7	3.77	15.3	15.8	-	14.6	13.8	-
2F	X-CrossNet [70]	TSE	19.9	20.5	-	-	-	-	14.6	14.1	-
2G	X-SepFormer [71]		18.9	19.1	3.74	-	-	-	-	-	-
2H	CIENet-mDPTNet [36]		21.4	21.6	3.91	16.6	17.0	2.70	15.7	14.3	2.55
2I	CIENet-C2F-mDPTNet [35]		21.9	22.1	3.94	17.3	17.6	2.77	17.5	16.0	2.72
2J	CIENet-Enh-mDPTNet [72]		21.5	21.8	-	17.2	17.5	-	17.2	15.8	-
2K	DCF-Net [73]		21.6	21.7	-	16.8	17.3	-	15.8	14.5	-
2L	USEF-TSE [37]		23.3	23.5	-	17.6	17.9	-	16.1	14.9	-
3	LExt-4s (TFGridNetV2)	TSE	24.1	24.3	4.10	18.3	18.6	2.97	18.3	16.7	2.94

Notes: Systems trained with dynamic mixing are marked by “DM”.

set, and therefore extending the enrollment length beyond their natural length often requires zero-padding, which adds no new speaker information. Shorter segments might have already contain sufficient cues, while longer ones mainly incur padding and computation.

For TFLocoFormerV1, the performance is saturated when the enrollment utterance is 2-second-long, and the model does not produce better performance when trained with longer enrollment utterances.

B. Prepending Vs. Appending Vs. Both

In Table II, we compare the results of three strategies for concatenating enrollment utterance with mixture. The first one prepends the enrollment utterance to the mixture, the second one appends the enrollment utterance to the mixture, while the third one prepends the first half of the enrollment utterance while appends the second half. In the third strategy, if the prepended segment is shorter than $E/2$, we pad zeros to its left, while if the segment concatenated on the right is shorter than $E/2$, we pad zeros to its right (i.e., the input structure is $[e_p; g; y; g; e_a]$, where e_p and e_a respectively denote the enrollment signals for prepending and appending). The rationale of doing so is to have the active speech in the enrollment utterance closer to the mixture. Each strategy is applied consistently for training and testing.

In Table II, we do not observe better performance by using the second and third strategies, especially for TFLocoFormerV1. This is possibly because, during training, a fixed-length segment is sampled from the original mixture and used for training, while, at run time, the input mixture is in its full length, which can be

different from the segment length used for model training. In this case, if the second and third strategies are used, the positional encodings of TF-LocoFormer used for the appended enrollment utterance would be different between training and testing, potentially creating difficulties for TF-LocoFormer to perform TSE. Therefore, we try a version without using positional encoding, denoted as “NoPE”, in Table II. However, we observe that, despite using the NoPE version, TFLocoFormerV1 still shows no significant improvement. We have also listed the TSE results on the validation set. By comparing the metrics on the validation set and the test set, we find that TFGGridNetV1 is more robust than TFLocoFormerV1, as the test scores of TFGGridNetV1 in Table II are better than its validation scores, while the test scores of TFLocoFormerV1 are worse than its validation scores.

From Table II, we observe that the use of the first strategy performs better than the other two for both TFGGridNetV1 and TFLocoFormerV1. Therefore, we adopt the first strategy in subsequent experiments. It should be noted that we only compared their performance when the enrollment length is 4 seconds, and their relative performance may change for other enrollment lengths.

C. V1 Vs. V2 Models of TF-GridNet and LocoFormer for LExt

In Table III, based on the WSJ0-2mix dataset, we compare the TSE performance of using TFGGridNetV1 and TFGGridNetV2, and TFLocoFormerV1 and TFLocoFormerV2 in LExt. We observe that the V2 models outperform V1, and TFGGridNet outperforms TFLocoFormer. We hence use TFGGridNetV2 and prepend 4-second enrollment utterance in default in the following experiments.

TABLE V
PERFORMANCE COMPARISON OF TSE FOR 2-SPEAKER MIXTURES WITH SAME AND DIFFERENT GENDERS. RESULTS ARE BASED ON WSJ0-2MIX.

System	SDR (dB)↑		PESQ↑	
	Different	Same	Different	Same
Mixture	2.5	2.7	2.29	2.34
SpeakerBeam [19]	12.0	6.9	2.82	2.43
SpEx [74]	19.3	14.7	3.53	3.16
SpEx+ [66]	19.3	14.7	3.53	3.16
SEF-Net [38]	20.2	18.7	3.56	3.48
USEF-TSE [37]	24.8	24.4	3.96	3.88
LExt (TFGridNetV2)	25.6	25.1	4.19	4.16

Notes: Following the setup in Table VII of the USEF-TSE paper [37], for each mixture in the test set, we only report the TSE result for the first speaker, which always has higher energy (this is why the mixture SDRs are much higher than 0 dB).

D. LExt Vs. Existing TSE Approaches

Table IV compares the result of LExt with the ones reported in previous studies, based on the WSJ0-2mix, WHAM!, and WHAMR! datasets. We observe that LExt obtains clearly better TSE performance than existing approaches on all the three datasets, although being very simple. Table IV additionally lists speaker separation results (denoted as “SS”) on the same datasets. We will discuss them later in Section V-G.

WSJ0-2mix [5] is an anechoic dataset designed for two-speaker separation. It does not contain environmental noises and room reverberation. Notice that prepending an enrollment utterance to the mixture is equivalent to having the target speaker talk first and longer. This onset information could help the DNN identify and extract the target speaker. From the TSE results reported in Table IV, we observe that LExt obtains state-of-the-art performance, reaching 24.1 dB SI-SDR. WHAMR! [43] is a noisy-reverberant dataset designed for two-speaker separation. Since the enrollment utterance is anechoic without any noises and reverberation, the resulting concatenated mixture would exhibit some discrepancies between the signals in the time range of the enrollment utterance and those in the time range of the original mixture signal. From the TSE results presented in Table IV, we find that LExt achieves strong performance, obtaining 18.3 dB SI-SDRi. This indicates the effectiveness of LExt at TSE in noisy-reverberant conditions and that LExt can deal with the discrepancies. We can draw similar conclusions based on the evaluation results on the WHAM! dataset [42]. Although, in Table IV, LExt only exploits 4-second active enrollment speech rather than the full-length enrollment, it already shows better performance than the competing algorithms, which exploit full-length enrollment. We also tried to use full-length enrollment for LExt, but the model is too slow to train in the TFGridNetV2 setup and we hence leave it for future investigation.

In Table V, based on the two-speaker mixtures in the WSJ0-2mix test set, we compare the performance of LExt and existing studies for mixtures consisting of same- and different-gender speaker signals. In both cases, we observe that LExt obtains better TSE performance than existing studies reporting the result in the same setup.

Fig. 2 plots the histograms of the SI-SDRi scores of USEF-TSE (in row 2 L of Table IV) and LExt (in row 3 of Table IV) on

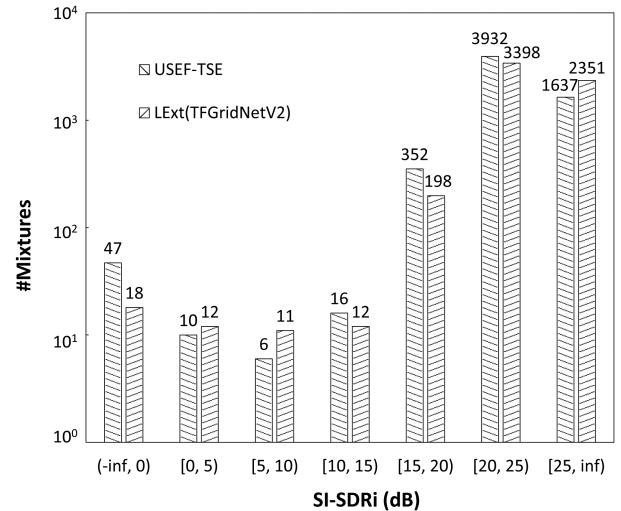


Fig. 2. Histogram of SI-SDRi scores on WSJ0-2mix test set. “Inf” denotes infinity.

the WSJ0-2mix test set. We consider the cases when the SI-SDRi score is less than 0 dB as failed cases (i.e., the left-most bar), and observe that LExt has fewer failed cases than USEF-TSE. In addition, we observe that LExt has more mixtures with more than 25.0 dB improvement in SI-SDR.

E. Sensitivity to Length of Enrollment Utterance

In Table VI, we compare the performance of LExt, a TSE system based on fixed-length speaker embeddings, and USEF-TSE (based on variable-length speaker embeddings) when trained with various enrollment utterance lengths. This comparison can show the sensitivity of various approaches to the length of enrollment utterances. It can reveal which approach needs the least amount of enrollment speech.

We leverage the implementation open-sourced in the WeSep toolkit [14] to build the TSE system based on fixed-length speaker embeddings. See the details in Section IV-E. For the USEF-TSE system, we leverage the USEF-TSE implementation⁶ released by the original authors [37]. The three systems in this experiment are all configured to use TFGridNetV1 as the speaker extraction network. They are trained using exactly the same training configuration, and the only difference is in the way speaker conditioning is applied.

TSE systems based on fixed-length speaker embeddings and USEF-TSE, in default, both use each enrollment utterance in its full length for training and evaluation. We denote this way as *Full without SAD*. In addition, after applying SAD to the enrollment utterances, for both training and testing we fix the enrollment length at E seconds, where E enumerates the set of $\{0.1, 0.25, 0.5, 1.0, 2.0, 4.0\}$, and report the results in Table VI.

From the results, we observe that for various lengths of enrollment utterances, LExt obtains consistently better TSE performance than the other two. In addition, LExt works reasonably well even if the enrollment utterance is as short as 0.5 seconds,

⁶Available at <https://github.com/ZBang/USEF-TSE>.

TABLE VI
SI-SDR_i (dB) COMPARISON OF LEXT VS. TSE WITH FIXED-LENGTH SPEAKER EMBEDDINGS AND USEF-TSE BASED ON WSJ0-2MIX

System	DNN arch.	Length of enrollment utterance (s)					
		with SAD				without SAD	
		0.1	0.25	0.5	1.0	2.0	4.0
TSE with fixed-length speaker embeddings [14]	TFGridNetV1	Failed	15.0	14.5	18.0	19.1	18.6
USEF-TSE (with variable-length speaker embeddings) [37]	TFGridNetV1	17.7	19.5	20.4	21.8	22.2	22.9
LExt	TFGridNetV1	18.7	20.8	22.0	22.3	22.8	23.0

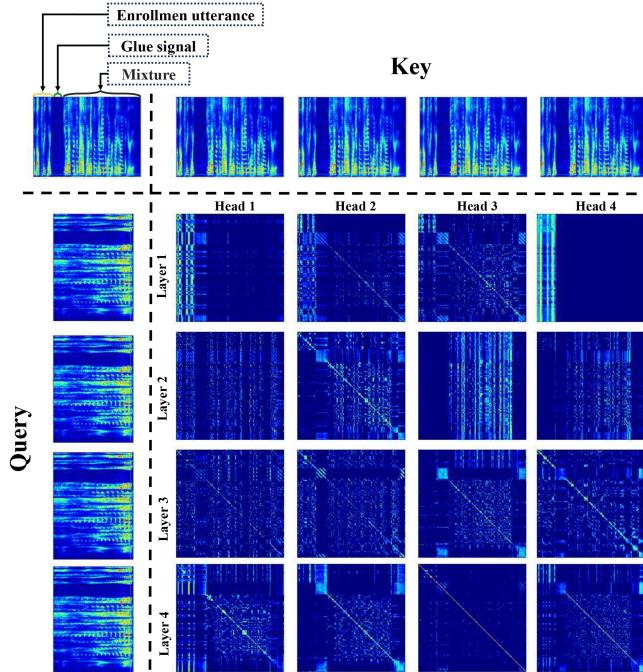


Fig. 3. Illustration of attention maps in TFGGridNetV1 based LExt reported in row 1 C of Table I. In the top-left sub-plot, we mark down the time ranges of the enrollment utterance, glue signal and mixture. Best viewed in color.

while in this case the TSE system based on fixed-length speaker embeddings performs significantly worse (i.e., 22.0 vs. 14.5 dB SI-SDR_i). The degradation of the TSE model based on fixed-speaker embedding could be attributed to its pooling operation to get fixed-length speaker embedding: when the enrollment is very short, the aggregated embedding may become unstable and less reliable. For USEF-TSE, the enrollment and mixture are fused only once at the input of the network. As the network goes deeper, the enrollment cues may gradually fade away, and this effect is exacerbated when the enrollment itself is very short. In contrast, LExt directly concatenates the enrollment with the mixture, so the speaker information is likely to remain salient throughout the layers. This design may be responsible for why LExt can maintain stronger TSE performance even in the extremely-short 0.5-second enrollment condition.

F. Illustration of Attention Maps

Fig. 3 visualizes the attention maps of the TFGGridNetV1 based LExt system reported in row 1 C of Table I, based on a test mixture in WSJ0-2mix. In TFGGridNetV1, there are 4 self-attention layers, each with 4 attention heads, and Fig. 3 plots all the 16

(= 4 × 4) attention maps. We observe that TFGGridNetV1 indeed leverages self-attention to exploit the speaker cues in the time range of the prepended enrollment utterance, mainly in the first self-attention layer.

G. TSE Vs. Speaker Separation

In datasets designed for two-speaker separation (e.g., WSJ0-mix, WHAM! and WHAMR!), it is commonly observed in many TSE studies [8] that first separating all the speakers and then, in an oracle way, aligning the speaker estimates to target speakers (denoted as SS, meaning “speaker separation”) can yield better estimation of the target speakers than applying TSE to extract each of the target speakers in turn. This can be observed in Table IV by, e.g., (a) comparing row 2E and 2 L with 1 L, where X-TF-GridNet [31] and USEF-TSE [37] designed for TSE are both not better than TF-GridNet [39] designed for SS; (b) comparing row 1 M and 2 F, where X-CrossNet [70] designed for TSE is not better than TF-CrossNet [65] designed for SS; and (c) comparing row 1 E with 2 G, where X-SepFormer [71] designed for TSE is worse than SepFormer designed for SS [40]. This indicates that many speaker conditioning mechanisms cannot sufficiently exploit speaker cues in enrollment utterances for TSE. Comparing row 3 (TF-GridNet and LExt based TSE) with 1 L (TF-GridNet based SS), we observe that LExt based TSE outperforms SS. This comparison indicates that LExt is a very effective mechanism at exploiting speaker cues in enrollment utterances for TSE.

H. Effects of Utilizing Glue Signal

To assess the contribution of the glue signal, we conduct an ablation study by removing it from the LExt input. Recall that the glue signal is introduced to provide an explicit temporal cue that can indicate the boundary between the enrollment and mixture. Since the enrollment is fixed in length and in this case the boundary information is already clear, it is important to verify whether this explicit temporal cue can still play a useful role. In addition, we investigate the impact of using different sample values for the glue signal. Since the glue signal is only used to delimit the enrollment utterance from the mixture signal, we set its sample values to 5.0. The rationale of setting them to 5.0 is because we perform gain normalization on both the enrollment and the mixture before feeding them to the network (see Section III-E), which means that the sample values of the enrollment and mixture (after gain normalization) will hardly reach 5.0, and therefore a signal with sample values set to 5.0 can be a very informative delimiter.

TABLE VII

ABLATION RESULTS OF LEXT WITH AND WITHOUT USING GLUE SIGNAL, AND USING DIFFERENT GLUE VALUES. EXPERIMENTS ARE BASED ON WSJ0-2MIX AND TFGRIDNETV1.

ID	Enrollment	Glue Value	Glue Length (ms)	SI-SDRi (dB)↑
1	Fixed at 1s	-	0.0	22.4
2	Fixed at 1s	0.0	32.0	22.3
3	Fixed at 1s	5.0	32.0	22.5
4	Full	-	0.0	20.5
5	Full	0.0	32.0	22.6
6	Full	5.0	32.0	23.0

TABLE VIII

EFFECTS OF SAD AT DIFFERENT ENROLLMENT LENGTHS. EXPERIMENTS ARE BASED ON WSJ0-2MIX AND TFGRIDNETV1. Δ = (SI-SDRi WITH SAD) – (SI-SDRi w/o SAD).

Length(s)	SI-SDRi (dB) w/o SAD	SI-SDRi (dB) with SAD	Δ
0.25	9.7	20.8	11.1
0.50	19.8	22.0	2.2
1.00	22.6	22.3	-0.3
2.00	22.5	22.8	0.2

The results are shown in Table VII. Firstly, we observe that setting glue values to 5.0 can produce better results, indicating that setting glue values to 5.0 can better serve as a delimiter between enrollment and mixture. Secondly, we find that the glue signal does not play a decisive role in the network's differentiation between enrollment and mixture. When we use a fixed-length enrollment, the network can still determine through training that the fixed-length signal at the front is the enrollment. However, when the length of the enrollment is not fixed and the original, full-length enrollment utterance is used, the presence of the glue signal enables the network to better distinguish between enrollment and mixture.

I. Effects of Applying SAD to Enrollment

We study how SAD would affect LExt across different enrollment lengths, including very short lengths (0.25 s and 0.5 s) and longer ones (1 s and 2 s). For each length, we compare models trained and evaluated without applying SAD and with applying SAD to the enrollment.

From the results in Table VIII, we observe that for short enrollments, when SAD is not used, there is a significant drop in TSE performance. Specifically, when the enrollment length is as short as 0.25 s, the performance of the “without SAD” setting drops drastically: its SI-SDRi value is only at 9.7 dB, while the “with SAD” setting achieves an SI-SDRi of 20.8 dB. This is because during testing, when we cut a short segment from the entire enrollment speech for LExt, without SAD, we could obtain a silent segment or a segment without too much active speech, which would lead to extraction failures. However, as the signal length increases, the performance with and without SAD becomes relatively close.

VI. DISCUSSIONS

This section describes several limitations of LExt compared with fixed-length speaker embedding based systems for TSE.

A. Limitations of LExt for Offline TSE

LExt increases the input length by concatenating the enrollment utterance with the mixture, which leads to higher computational cost during TSE. The amount of increased computation is in proportion to the length of the concatenated enrollment utterance. However, this may not be a serious problem, as in our experiments we find that an enrollment utterance as short as 0.25 or 0.5 seconds (see Table I) can already produce very strong extraction performance. This length is much shorter than that of typical mixture signals, which is typically multi-seconds long.

Another related limitation is that every time LExt is used for inference, the enrollment utterance has to be processed together with the mixture signal. We cannot store the processing results of the enrollment utterance on hard drives for later use, as they are different for different mixture signals. This is a weakness compared to TSE systems based on fixed-length speaker embeddings, where a speaker embedding is extracted only once for each speaker and can be stored on disks, or in cache, for later use.

B. Limitations of LExt for Real-Time, Frame-Online TSE

Another limitation may emerge when applying LExt for real-time, frame-online TSE, since, during inference, the DNN needs to refer to the enrollment utterance concatenated at the very beginning of the mixture for TSE. In real-time scenarios, where the computation resource is often limited, referring to the tensors computed based on the enrollment utterance by using, e.g., attention mechanisms would be costly. Future research will explore solutions to this problem.

On the other hand, in online TSE, we can cache the tensors computed from the enrollment utterance (assuming only using prepending in LExt), and re-use them to extract the target speaker in different mixtures. That is, the second limitation described in Section VI-A can be avoided in online TSE.

VII. CONCLUSION

We have proposed LExt for TSE. By concatenating an enrollment utterance to the mixture to create an artificial onset for the target speaker, LExt is found highly-effective at extracting the target speaker, achieving state-of-the-art TSE performance on multiple widely-adopted datasets. Even if the enrollment utterance used for concatenation is as short as 0.25 or 0.5 seconds, LExt can still obtain reasonably strong TSE performance. LExt is extremely simple to implement and does not require additional design of DNN architectures for speaker conditioning. Moving forward, we plan to extend LExt for related target-speaker speech processing tasks and address its limitations described in Section VI.

In closing, we highlight that, in the past decade, TSE has been conducted mainly following the convention of designing additional DNN modules for speaker conditioning (e.g., by leveraging fixed- or variable-length speaker embeddings) so that the speaker extraction network can learn to extract the target speaker. LExt, for the first time, proposes to use no additional speaker-conditioning modules, and the speaker

extraction network can learn to leverage onset cues for TSE. This approach, we think, could potentially motivate the design of many TSE algorithms in future research.

ACKNOWLEDGMENT

This work was done while the author Pengjie Shen was Visiting Student at Southern University of Science and Technology.

REFERENCES

- [1] C. E. Cherry, "Some experiments on the recognition of speech, with one and with two ears," *J. Acoustical Soc. Amer.*, vol. 25, no. 5, pp. 975–979, 1953.
- [2] J. H. McDermott, "The cocktail party problem," *Curr. Biol.*, vol. 19, no. 22, pp. 1024–1027, 2009.
- [3] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 10, pp. 1702–1726, Oct. 2018.
- [4] M. Kolbaek, D. Yu, Z.-H. Tan, and J. Jensen, "Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 10, pp. 1901–1913, Oct. 2017.
- [5] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2016, pp. 31–35.
- [6] M. Delcroix et al., "Improving speaker discrimination of target speech extraction with time-domain speakerbeam," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 691–695.
- [7] M. Delcroix, J. B. Vázquez, T. Ochiai, K. Kinoshita, Y. Ohishi, and S. Araki, "SoundBeam: Target sound extraction conditioned on sound-class labels and enrollment clues for increased performance and continuous learning," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 31, pp. 121–136, 2023.
- [8] M. Elmishawwi, W. Mack, S. Chakrabarty, and E. A. P. Habets, "New insights on target speaker extraction," 2022, *arXiv:2202.00733*.
- [9] M. Ge, C. Xu, L. Wang, E. S. Chng, J. Dang, and H. Li, "L-SpEx: Localized target speaker extraction," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2022, pp. 7287–7291.
- [10] D. Michelsanti et al., "An overview of deep-learning-based audio-visual speech enhancement and separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 1368–1396, 2021.
- [11] T. Ochiai et al., "Target sound information extraction: Speech and audio processing with neural networks conditioned on target clues," *Acoustical Sci. Technol.*, vol. 46, pp. 197–209, 2025.
- [12] J. Wang et al., "Deep extractor network for target speaker recovery from single channel speech mixtures," in *Proc. Interspeech*, 2018, pp. 307–311.
- [13] Q. Wang et al., "VoiceFilter: Targeted voice separation by speaker-conditioned spectrogram masking," in *Proc. Interspeech*, 2019, pp. 2728–2732.
- [14] S. Wang et al., "WeSep: A scalable and flexible toolkit towards generalizable target speaker extraction," in *Proc. Interspeech*, 2024, pp. 4273–4277.
- [15] Z. Wang et al., "The multimodal information based speech processing (MISP) 2022 challenge: Audio-visual diarization and recognition," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2023, pp. 1–5.
- [16] S. Wu et al., "The multimodal information based speech processing (MISP) 2023 challenge: Audio-visual target speaker extraction," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2024, pp. 8351–8355.
- [17] K. Žmolíková, M. Delcroix, T. Ochiai, K. Kinoshita, J. Cernocký, and D. Yu, "Neural target speech extraction: An overview," *IEEE Signal Process. Mag.*, vol. 40, no. 3, pp. 8–29, May 2023.
- [18] K. Žmolíková, M. Delcroix, K. Kinoshita, T. Higuchi, A. Ogawa, and T. Nakatani, "Speaker-aware neural network based beamformer for speaker extraction in speech mixtures," in *Proc. Interspeech*, 2017, pp. 2655–2659.
- [19] K. Žmolíková et al., "SpeakerBeam: Speaker aware neural network for target speaker extraction in speech mixtures," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 4, pp. 800–814, Aug. 2019.
- [20] A. Ephrat et al., "Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation," *ACM Trans. Graph.*, vol. 37, no. 4, Apr. 2018, Art. no. 112.
- [21] A. Gabbay, A. Shamir, and S. Peleg, "Visual speech enhancement," in *Proc. Interspeech*, 2018, pp. 1170–1174.
- [22] H. Wang, X. Zhang, and D. Wang, "Fusing bone-conduction and air-conduction sensors for complex-domain speech enhancement," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 30, pp. 3134–3143, 2022.
- [23] M. Delcroix, K. Žmolíková, T. Ochiai, K. Kinoshita, and T. Nakatani, "Speaker activity driven neural speech extraction," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2021, pp. 6099–6103.
- [24] R. Gu, S.-X. Zhang, Y. Xu, L. Chen, Y. Zou, and D. Yu, "Multi-modal multi-channel target speech separation," *IEEE J. Sel. Topics Signal Process.*, vol. 14, no. 3, pp. 530–541, Mar. 2020.
- [25] E. Ceolini et al., "Brain-informed speech separation (BISS) for enhancement of target speaker in multitalker speech perception," *NeuroImage*, vol. 223, 2020, Art. no. 117282.
- [26] Z. Pan, M. Borsdorff, S. Cai, T. Schultz, and H. Li, "NeuroHeed: Neuro-steered speaker extraction using EEG signals," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 32, pp. 4456–4470, 2024.
- [27] T. Afouras, J. S. Chung, and A. Zisserman, "My lips are concealed: Audio-visual speech enhancement through obstructions," in *Proc. Interspeech*, 2019, pp. 4295–4299.
- [28] T. Ochiai, M. Delcroix, K. Kinoshita, A. Ogawa, and T. Nakatani, "Multi-modal SpeakerBeam: Single channel target speech extraction with audio-visual speaker clues," in *Proc. Interspeech*, 2019, pp. 2718–2722.
- [29] H. Sato, T. Ochiai, K. Kinoshita, M. Delcroix, T. Nakatani, and S. Araki, "Multimodal attention fusion for target speaker extraction," in *Proc. IEEE Spoken Lang. Technol. Workshop*, 2021, pp. 778–784.
- [30] S. Cornell, Z.-Q. Wang, Y. Masuyama, S. Watanabe, M. Pariente, and N. Ono, "Multi-channel target speaker extraction with refinement: The WAVLAB submission to the second clarity enhancement challenge," in *Proc. Clarity*, 2022, pp. 1–3.
- [31] F. Hao, X. Li, and C. Zheng, "X-TF-GridNet: A time-frequency domain target speaker extraction network with adaptive speaker embedding fusion," *Inf. Fusion*, vol. 112, 2024, Art. no. 102550.
- [32] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2018, pp. 5329–5333.
- [33] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," in *Proc. Interspeech*, 2020, pp. 3830–3834.
- [34] X. Xiao et al., "Single-channel speech extraction using speaker inventory and attention network," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 86–90.
- [35] X. Yang, C. Bao, and X. Chen, "Coarse-to-fine target speaker extraction based on contextual information exploitation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 32, pp. 3795–3810, 2024.
- [36] X. Yang, C. Bao, J. Zhou, and X. Chen, "Target speaker extraction by directly exploiting contextual information in the time-frequency domain," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2024, pp. 10476–10480.
- [37] B. Zeng and M. Li, "USEF-TSE: Universal speaker embedding free target speaker extraction," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 33, pp. 2110–2124, 2025.
- [38] B. Zeng, H. Suo, Y. Wan, and M. Li, "SEF-Net: Speaker embedding free target speaker extraction network," in *Proc. Interspeech*, 2023, pp. 3452–3456.
- [39] Z.-Q. Wang, S. Cornell, S. Choi, Y. Lee, B.-Y. Kim, and S. Watanabe, "TF-GridNet: Integrating full- and sub-band modeling for speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 31, pp. 3221–3236, 2023.
- [40] C. Subakan, M. Ravanelli, S. Cornell, M. Bronzi, and J. Zhong, "Attention is all you need in speech separation," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2021, pp. 21–25.
- [41] K. Saijo, G. Wichern, F. G. Germain, Z. Pan, and J. L. Roux, "TF-LocoFormer: Transformer with local modeling by convolution for speech separation and enhancement," in *Proc. Int. Workshop Acoustic Signal Enhancement*, 2024, pp. 205–209.
- [42] G. Wichern et al., "WHAM!: Extending speech separation to noisy environments," in *Proc. Interspeech*, 2019, pp. 1368–1372.
- [43] M. Maciejewski et al., "WHAMR!: Noisy and reverberant single-channel speech separation," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 696–700.
- [44] H. Taherian, V. A. Kalkhorani, A. Pandey, D. Wong, B. Xu, and D. Wang, "Towards explainable monaural speaker separation with auditory-based training," in *Proc. Interspeech*, 2024, pp. 572–576.
- [45] A. Pandey and D. Wang, "Attentive training: A new training framework for speech enhancement," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 31, pp. 1360–1370, 2023.

- [46] S. Cornell et al., "Multi-channel speaker extraction with adversarial training: The WAVLAB submission to the clarity ICASSP 2023 grand challenge," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2023, pp. 1–2.
- [47] S. He, W. Xue, Y. Yang, H. Zhang, J. Pan, and X. Zhang, "Enhancing target speaker extraction with hierarchical speaker representation learning," *Neural Netw.*, vol. 188, 2025, Art. no. 107388.
- [48] Y. Hao, J. Xu, P. Zhang, and B. Xu, "WASE: Learning when to attend for speaker extraction in cocktail party environments," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2021, pp. 6104–6108.
- [49] N. Kanda, Y. Gaur, X. Wang, Z. Meng, and T. Yoshioka, "Serialized output training for end-to-end overlapped speech recognition," in *Proc. Interspeech*, 2020, pp. 2797–2801.
- [50] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "SDR - half-baked or well done?," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 626–630.
- [51] K. Tan and D. Wang, "Learning complex spectral mapping with gated convolutional recurrent networks for monaural speech enhancement," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 380–390, 2020.
- [52] Z.-Q. Wang, P. Wang, and D. Wang, "Complex spectral mapping for single- and multi-channel speech enhancement and robust ASR," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 1778–1787, 2020.
- [53] Z.-Q. Wang, P. Wang, and D. Wang, "Multi-microphone complex spectral mapping for utterance-wise and continuous speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 2001–2014, 2021.
- [54] A. Courville et al., *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [55] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 4, pp. 1462–1469, Jul. 2006.
- [56] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, "Perceptual evaluation of speech quality (PESQ)-A new method for speech quality assessment of telephone networks and codecs," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, vol. 2, 2001, pp. 749–752.
- [57] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing ideal time-frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, Aug. 2019.
- [58] Y. Luo, Z. Chen, and T. Yoshioka, "Dual-path RNN: Efficient long sequence modeling for time-domain single-channel speech separation," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 46–50.
- [59] N. Zeghidour and D. Grangier, "Wavesplit: End-to-end speech separation by speaker clustering," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 2840–2849, 2021.
- [60] J. Rixen and M. Renz, "QDPN-quasi-dual-path network for single-channel speech separation," in *Proc. Interspeech*, 2022, pp. 5353–5357.
- [61] J. Chen, Q. Mao, and D. Liu, "Dual-path transformer network: Direct context-aware modeling for end-to-end monaural speech separation," in *Proc. Interspeech*, 2020, pp. 2642–2646.
- [62] S. Zhao and B. Ma, "MossFormer: Pushing the performance limit of monaural speech separation using gated single-head transformer with convolution-augmented joint self-attentions," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2023.
- [63] S. Zhao et al., "MossFormer2: Combining transformer and RNN-Free recurrent network for enhanced time-domain monaural speech separation," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2024, pp. 10356–10360.
- [64] W. Ravenscroft, S. Goetze, and T. Hain, "On time domain conformer models for monaural speech separation in noisy reverberant acoustic environments," in *Proc. IEEE Autom. Speech Recognit. Understanding Workshop*, 2023, pp. 1–7.
- [65] V. A. Kalkhorani and D. Wang, "TF-CrossNet: Leveraging global, cross-band, narrow-band, and positional encoding for single- and multi-channel speaker separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 32, pp. 4999–5009, 2024.
- [66] 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, pp. 1406–1410.
- [67] C. Deng et al., "Robust speaker extraction network based on iterative refined adaptation," in *Proc. Interspeech*, 2021, pp. 3530–3534.
- [68] M. Ge, C. Xu, L. Wang, E. S. Chng, J. Dang, and H. Li, "Multi-stage speaker extraction with utterance and frame-level reference signals," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2021, pp. 6109–6113.
- [69] L. Yang, W. Liu, L. Tan, J. Yang, and H. G. Moon, "Target speaker extraction with ultra-short reference speech by VE-VE framework," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2023, pp. 1–5.
- [70] C. Sun and B. Qin, "X-CrossNet: A complex spectral mapping approach to target speaker extraction with cross attention speaker embedding fusion," 2024, *arXiv:2411.13811*.
- [71] K. Liu, Z. Du, X. Wan, and H. Zhou, "X-SepFormer: End-to-End speaker extraction network with explicit optimization on speaker confusion," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2023, pp. 1–5.
- [72] X. Yang, C. Bao, X. Zhang, and X. Chen, "Target speaker extraction method by emphasizing the active speech with an additional enhancer," in *Proc. Asia Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2024, pp. 1–6.
- [73] K. Xue, R. Fan, C. Sun, P. Zhao, and J. An, "DCF-Net: Efficient target speaker extraction by leveraging mixture and enrollment interactions," *IEEE Signal Process. Lett.*, vol. 32, pp. 3240–3244, 2025.
- [74] C. Xu, W. Rao, E. S. Chng, and H. Li, "SpEx: Multi-scale time domain speaker extraction network," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 1370–1384, 2020.