System Description

Participants: Jiayang Li, Fei Zhao

Team name: Team-Fly

1. System Overview:

This system is a speech separation system for Target Speaker Extraction (TSE), based on a causal SE-SSE model framework. The system takes noisy/mixed speech and the enrollment utterance of the target speaker as inputs, and outputs the extracted target speaker speech from the mixture.

2. Model Framework:

The overall model framework of this system is based on the Band-Split Recurrent Neural Network (BSRNN), combined with target-speaker conditioning information to achieve target speaker extraction.

2.1. Time-Frequency Feature Analysis:

The system first applies a complex STFT transform to both the mixed speech and the target speaker enrollment speech, converting the waveforms into the time-frequency domain. The model uses a sampling rate of 16 kHz, an STFT window length of 512, and a frame shift of 256.

2.2. Shared Spectral Feature Encoder:

The mixed speech spectrum and the enrollment speech spectrum are processed by the same shared convolutional encoder for feature extraction. The encoder consists of three two-dimensional convolution layers, normalization layers, and PReLU activation functions. In causal mode, the model uses causal layer normalization and applies padding only from historical frames along the time dimension, thereby supporting online/streaming processing.

2.3. Subband Feature Representation:

The encoded spectral features are divided into multiple frequency subbands following the BSRNN design. Each subband is independently normalized and projected to obtain a fixed-dimensional subband representation. In this system, the subband feature dimension is 128, and the BSRNN subband partition structure is derived from the 512-point STFT configuration.

2.4. Target Speaker Embedding Extraction:

The system extracts subband-level target speaker information from the enrollment speech. Specifically, the enrollment speech is processed by the shared encoder and subband splitting

module, and then fed into a lightweight CNN-based speaker embedding extractor. This module contains one-dimensional convolution layers, Attentive Statistics Pooling (ASP), and fully connected projection layers, generating a 128-dimensional speaker embedding for each subband. This design enables the model to directly use the enrollment speech to extract target speaker information during the forward pass, rather than relying only on an external fixed speaker vector.

2.5. Speaker-Conditioned Separation:

The target speaker subband embeddings are fused with the mixed speech subband features through “multiplicative modulation + residual connection”. The fused features are then fed into the BSRNN separator. The BSRNN separator consists of multiple repeated modules for intra-subband temporal modeling and cross-subband communication. In this system configuration, the number of repeats is 6. In causal mode, the temporal RNN modeling uses a unidirectional structure to satisfy online processing requirements while preserving cross-frequency-band information exchange.

2.6. Complex Mask Estimation and Waveform Reconstruction:

The output of the BSRNN separator is passed to the BandMasker to generate complex masks for each subband. The complex masks are applied to the mixture STFT to obtain the estimated complex spectrum of the target speaker. Finally, iSTFT is used to reconstruct the time-domain waveform. The final output of the model is the estimated speech of the target speaker.

3. Causality and Algorithmic Latency

Although the neural network components are designed to be causal, the complete system still has algorithmic latency introduced by the frame-based STFT/iSTFT analysis-synthesis procedure.

The model uses a 512-sample STFT window and a 256-sample frame shift at a sampling rate of 16 kHz. The only intentional future dependency comes from the centered STFT analysis window. Therefore, the theoretical algorithmic look-ahead is:

Theoretical algorithmic latency

= STFT window length / 2

= 512 / 2 samples

= 256 samples

= 16 ms (16 kHz)

No additional look-ahead is introduced by the neural network separator. Specifically:

- The shared convolutional encoder uses causal padding along the time dimension.

- The temporal RNN in the BSRNN separator is unidirectional in causal mode.

- Cross-band communication is performed across frequency bands and does not access future time frames.

- No extra chunk accumulation or output buffering is used beyond the STFT/iSTFT frame-based processing.

Therefore, the nominal theoretical algorithmic latency of the submitted system is 16 ms.

4. Official Validation Script Latency Verification

We further verified the causality and latency consistency of the submitted causal BSRNN Plan2 system using the official latency validation script. The validation was performed with the submitted checkpoint and the default perturbation-based verification setting.

The script reports the following future-dependency estimates:

Perturbation Time	Response Time	Estimated Future Dependency
1.000 s	0.977 s	22.7 ms
2.000 s	1.984 s	15.6 ms
1.234 s	1.217 s	17.0 ms
2.346 s	2.322 s	24.2 ms
3.459 s	3.441 s	17.8 ms

Summary:

Minimum future dependency: 15.6 ms

Mean future dependency: 19.5 ms

Maximum future dependency: 24.2 ms

The measured values are consistent with the theoretical latency of 16 ms. The small variation between the theoretical latency and the perturbation-based measurements is expected for a frame-based STFT/iSTFT system, since the detected response time depends on the perturbation position relative to the STFT frame grid, the 256-sample hop size, numerical precision, and the detection threshold used by the validation script. The maximum verified future dependency is 24.2 ms, which is within one frame-shift interval from the nominal 16 ms theoretical latency.

Thus, the submitted Track 1 system is causal with a theoretical algorithmic latency of 16 ms, and the official validation script verifies a maximum future-dependency estimate of 24.2 ms.

5. Training Setup:

The training objective of this system is the SI-SNR loss. The optimizer is Adam, with a learning rate of 1e-4 and a weight decay of 1e-5. A constant learning-rate scheduler is used. The training batch size is 8, the number of data loading workers is 8, and the maximum number of training epochs is 100. Distributed PyTorch training is launched on the V100 GPU partition, configured with 1 node and 8 GPUs per node.

6. Data Usage:

The training data is constructed following the voicefilter/TSE data generation method. The training data includes lists of clean speech, interfering speech, noise, room impulse responses (RIRs), and target speaker enrollment speech. During training, mixed speech is dynamically generated: target speaker speech is randomly selected, and 0, 1, or 2 interfering speakers are added according to the configured distribution. Noise is also added, and room impulse responses are applied. The energy relationship between the target speech and the interference/noise is controlled by sampled SIR and SNR values. The duration of training utterances is randomly sampled between 3 and 10 seconds.

The data sources used in this system include:

- TTS speech data

- AISHELL speech data

- DataTang speech data

- FreeST speech data

- Lenovo: a private 40-speaker dataset