The NPU-TEA System for the CHiME-8 NOTSOFAR-1 Challenge

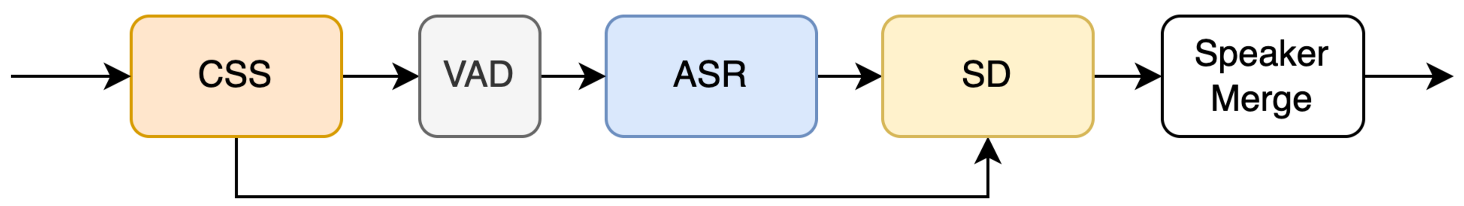
Kaixun Huang[1,2], Yue Li[1,2], Ziqian Wang[1,2], Hongji Wang[2], Wei Rao[2], Zhaokai Sun[1,2],
Zhiyuan Tang[2], Shen Huang[2], Yannan Wang[2], Tao Yu[2], Lei Xie[1], Shidong Shang[2]
[1]Audio, Speech and Language Processing Group (ASLP@NPU), Northwestern Polytechnical University, Xi'an, China
[2]Tencent Ethereal Audio Lab, Tencent Corporation, Shenzhen, China

Abstract

- Follow the baseline framework and include three main modules: CSS, ASR, and SD
- Enhanced the CSS module by integrating WavLM base plus model
- Use AdaLoRA to finetune the Whisper large-v2 as the ASR model
- Replaced the speaker embedding extraction model in the SD module with ResNet293 and Ecapa1024 (WavLM Large Frontend)
- Propose a combined Rover strategy to perform fusion on recognition results that include speaker labels
- TcpWER decreases by 37.65% for single-channel data and 32.11% for multi-channel data
- The submitted systems achieve 2nd place in both the single-channel and multi-channel tracks

System Overview

- The audio is separated into three non-overlapping audio using CSS
- Silence segments are removed from the audio using VAD
- Use the ASR module for recognition
- Use SD module to assign speakers and merge speakers with high similarity



Continuous Speech Separation

- We adopt conformer for speech separation in multi-channel track and conformer with WavLM for single-channel track
- Add results of the non-separation and the 2-channel separation to perform Rover
- Non-separation system can achieve better results than the separation system in single-channel data

Automatic Speech Recognition

Data Preparation

- Using the CSS model to separate audio into three tracks
- Use Whisper-large v2 for recognition, keep the audio closest to the ground truth
- Trim the extra words at the beginning and end of the ground truth compared to the prediction
- Keep the resulting audio-text pairs as training data

Training

- Use Whisper-large v2 as the ASR model and finetune it using AdaLoRA
- Use the fine-tuned ASR model to iterate the data Preparation again
- Build a 3-gram language model using NOTSOFAR and AMI datasets

Inference

- Remove silence from audio using Silero VAD
- Use ASR model to transcribe audio and return n-best results
- Rescore n-best results using a language model

Speaker Diarization

System configure

- Follow the NeMo diarization baseline and adopt the post-SR configuration
- Use the ResNet293 and Ecapa-tdnn1024 as speaker model

Training and Fine-tuning

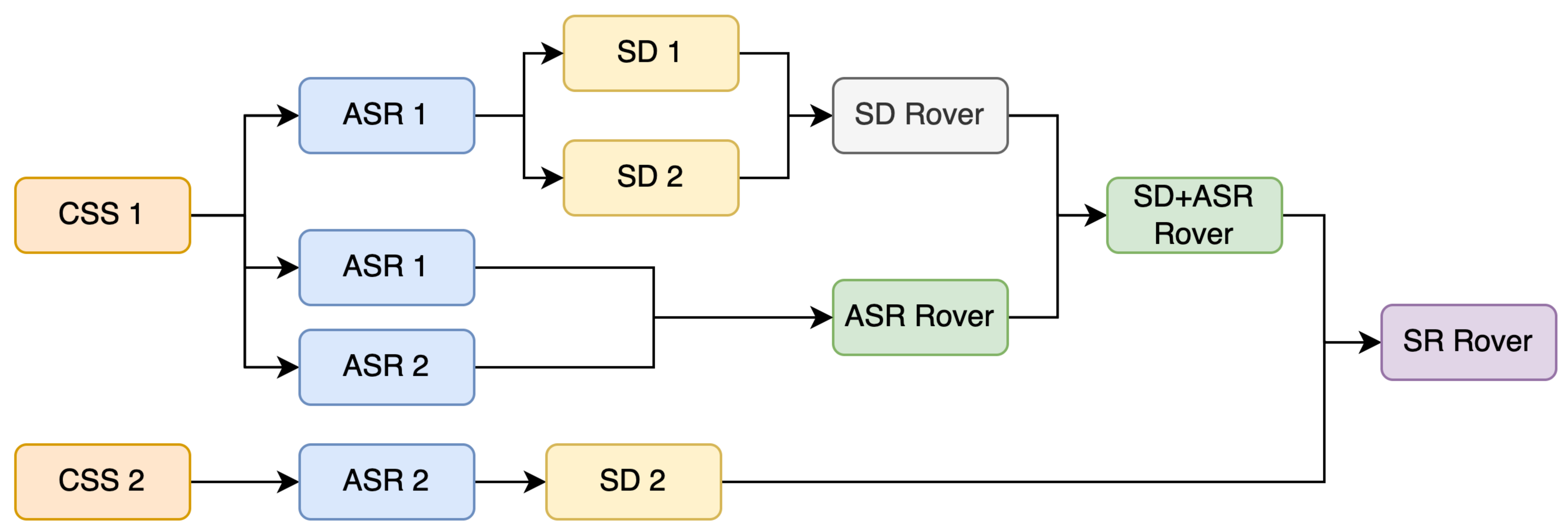
- Train with VoxCeleb 1&2, LibriSpeech, and WSJ, finetune with NOTSOFAR and AMI
- Apply noise, reverberation, and speed perturbation for data augmentation

Post-processing

- compute the speaker centers of all clustered speakers, merge speakers with cosine similarity above the threshold

System Fusion

- **ASR-Rover:** Apply Rover to the results of different ASR models using the same CSS
- **SD-Rover:** Apply DOVER-Lap to the results of different SD models using the same CSS
- **SR-Rover:** Match speakers across different systems, then apply the Rover algorithm to the same speakers
- **Combined Rover:** Merge ASR-Rover and SD-Rover results, then fuse with the best single-system result using SR-Rover



Experiments

Table 1: *Performance comparison among differnet speaker models tested on Vox1-O. The default ResNet293 model uses TSTP pooling layer with AAM-softmax loss, while ResNet293* uses MQMHASTP pooling layer with AAM-softmax intertopk-subcenter loss.*

No	Model	Training+Finetuning Data	EER(%)	minDCF
sd_titanet	TitaNet-L [4]	Vox1&2+SRE+Fisher+SWBD+LibriSpeech	0.68	0.087
sd_resnet1	ResNet293	TRAIN+AMI+NOTSOFAR-MC-CH0&CSS	0.399	0.035
sd_resnet2	ResNet293	TRAIN+AMI+NOTSOFAR-MC&SC-CH0&CSS	0.399	0.036
sd_resnet3	ResNet293	TRAIN+AMI+NOTSOFAR-MC&SC-CH0&CSSWavLM	0.399	0.036
sd_resnet4	ResNet293*	TRAIN+AMI+NOTSOFAR-MC&SC-CH0&CSS	0.404	0.030
sd_ecapa	Ecapa-tdnn1024	TRAIN+AMI+NOTSOFAR-MC&SC-CH0&CSS&CSSWavLM	0.441	0.062

- ResNet293 models significantly outperform the baseline TitaNet-L model
- By using the WavLM Large as the frontend, the Ecapa-tdnn1024 model also achieves a promising result

Table 2: *Results of all single and fused systems in both SC and MC tracks.*

Track	No	System	Dev-set-2 tcpWER (%)	Dev-set-2 tcORC WER (%)	Submission
SC	Baseline	css + Whisper large-v2 + Titanet-L	45.84	38.60	
	A1	css + asr + sd_resnet1	36.56	34.60	
	A2	css + asr + sd_resnet2	35.43	34.74	
	A3	css + asr_ngram + sd_resnet2	35.80	34.89	
	B1	css_wavlm + asr + sd_resnet3	33.10	30.14	
	B2	css_wavlm + asr + sd_resnet4	**32.80**	**29.91**	**sys1**
	B3	css_wavlm + asr_ngram + sd_resnet3	33.20	30.03	
	B4	css_wavlm + asr_ngram + sd_resnet4	33.11	29.99	
	B5	css_wavlm + asr_simu + sd_resnet4	33.13	29.84	
	B6	css_wavlm + asr_simu + sd_resnet3	33.44	29.73	
	C1	css_wavlm_2spk + asr + sd_resnet3	33.70	29.02	
	D1	wo_css + asr + sd_resnet2	**32.29**	**26.77**	**sys2**
	F1	A1 ∼ D1 rover1	32.93	28.95	
	F2	B1 ∼ D1 rover1	**29.77**	**30.02**	**sys3**
	F3	A1 ∼ D1 rover2	**28.58**	**28.94**	**sys4**
	F4	B1 ∼ D1 rover2	30.34	30.32	
MC	Baseline	css + Whisper large-v2 + Titanet-L	31.55	26.59	
	A1	css + asr + sd_resnet1	22.36	20.47	
	A2	css + asr + sd_resnet2	22.17	20.50	
	A3	css + asr_ngram + sd_titanet	22.87	20.50	
	A4	css + asr_ngram + sd_resnet2	**21.80**	**20.46**	**sys1**
	A5	css + asr_ngram + sd_resnet4	21.94	20.46	
	A6	css + asr_ngram + sd_ecapa	22.23	20.47	
	A7	css + asr_simu + sd_resnet1	22.52	20.75	
	A8	css + asr_ssl_ngram + sd_resnet2	**22.04**	**20.46**	**sys2**
	F1	A1 ∼ A7 rover1	**21.42**	**26.95**	**sys3**
	F2	A1 ∼ A7 rover2	**21.78**	**26.99**	**sys4**

- In single-channel data, the non-separation system performs better for audio with shorter overlapping times
- The best single-channel Rover result, F3, achieved 28.58% tcpWER, a 37.65% relative reduction
- The best multi-channel Rover result, F1, achieved 21.42% tcpWER, a 32.11% relative reduction