# Description of the Submission System for Team T021

anonymous

anonymous
{anonymous}@anonymous.com

## Abstract

This paper describes our work in CNSRC 2022 challenge. There are two tasks included in this challenge, speaker verification (SV) and speaker retrieval (SR). Task 1 (SV) involves two tracks: fixed track and open track, task 2 (SR)involves only open track. Our work focuses on open track of challenge SV. To get lower minDCF and EER, we combine five distinct subsystems, which are i-vector, x-vector, ECAPA-TDNN[1] with 80-dimensions FBANK, ECAPA-TDNN with 80-dimensions PCEN[2], and pre-trained WavLM[3]. CN-Celeb.T[4, 5] datasets are used in our training/finetuning process. Our experimental results show that subsystems with large differences achieve higher performance gains than those with tiny differences.

## 1. Data

CN-Celeb.T datasets are used in our experiments for all subsystems, except the pre-training of WavLM. This dataset contains 2793 speakers and 632740 utterance in about 1.3k hours of data. In WavLM, the pre-trained models using an 94k hours of English dataset, which consists of 60k hours of Libri-Light[6], 10k hours of GigaSpeech[7] and 24k hours of VoxPopuli[8].

## 2. Models

We use kaldi and pytorch to train our generative and discriminative models, Specifically, kaldi for i-vector and x-vector models, and pytorch for ECAPA-TDNN training and WavLM training and finetuning. All scores are normalized with Z-norm(zero score normalization). The final score file is the weighted average of all subsystems, and the weight value of each subsystem is 0.35 for WavLM, 0.1 for PCEN, 0.15 for FBANK, 0.25 for x-vector and 0.15 for i-vector respectively.

### 2.1. i-vector system

I-vector system is the only generative model compared to others. In our work, we simply adopt the kaldi baseline as our i-vector system, this is a traditional method using 24-dimensions MFCC with frame length 25ms and frame shift 10ms, energy-based VAD, Cepstral delta(Delta) and Cepstral Mean and Variance Normalization(CMVN) are performed in this system. Then we train an UBM with 2048 Gaussian component and a 400-dimensional i-vector. The back-ends consists of a LDA to decrease the dimensionality to 150, and a PLDA to get discriminative scores.

### 2.2. x-vector system

Same as i-vector system, x-vector system choose kaldi as backbone. Acoustic features are 30-dimensions MFCC with frame length 25ms and frame shift 10ms. We perform offline data augmentation using non-speech audio selected from MUSAN[9] and RIR_NOISES[10] during training. Then we combine the cleaned and augmented datasets at a ratio of 1:1. The network architecture is the standard TDNN, which output a 512-dimensional embedding, then LDA and PLDA are applied in the same way as i-vector.

### 2.3. ECAPA-TDNN FBANK system

ECAPA-TDNN achieves good performance across most speaker verification tasks. We trid this model during the competition for a better result of a single system. Differ from kaldi's povey window, we applied a hamming window and got 80-dimensions FBANK with frame length 25ms and frame shift 10ms. No VAD is adopted and online data reveberation, noisy, speed finetuning in time domain and SpecAugment[11] in frequency domain are applied while we training this models. We adopted the OneCycle[12] learning rate policy and aamsoftmax loss to train our system. Averaging of weights over different epoch/step of a same model are also adopted in this system. The model outputs a 192-dimensional embedding for backends processing. Besides, we also tried modify some layers but they did not work very well, for example, we replace the convolution kernel with an attention-based convolution kernel, which illustrated in Fig. 1. For the backends system, we use cosine similarity to measure the distance between two embeddings instead of LDA and PLDA. The results are presented in the following section.

### 2.4. ECAPA-TDNN PCEN system

The only difference between this model and the FBANK model is the acoustic features, and we choose PCEN with trainable parameters for the experiment.

### 2.5. pre-trained WavLM system

WavLM is a Large-Scale Self-Supervised Pre-Training model for Full Stack Speech Processing. It is aimed to learn a universal representions for all speech tasks. We choose the WavLM Base+ as our upstream model to extract features for downstream SV task. In our work, we adopt ECAPA-TDNN as the downstream model. We fixed the parameters of the WavLM Base+ model and trained the parameters of the downstream model. Specifically, we firstly get 13 embeddings of the WavLM Base+

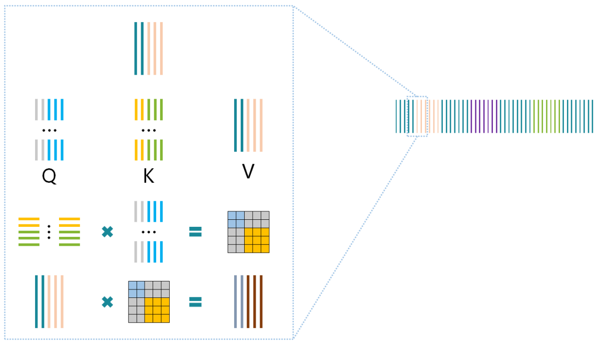

Figure 1: Description of our attention-based convolution kernel.

model's output, then we use a weighted sum layer to get a feature matrix which is the weighted sum of the 13 embeddings. The parameters of the weighted sum layer are trainable. Finally we use the feature matrix as the input of the downstream ECAPA-TDNN model on CN-Celeb.T datasets.

## 3. Results

Our final minDCF and EER results on the leaderboard are 0.4339 and 6.7920% respectively, which is the weighted average of all subsystems. The results of our models are shown in the Table 1. Our results show that subsystems with large differences(fusion 1&2, fusion 1&4, fusion 3&5) achieve higher performance gains than those with tiny differences(fusion 3&4).

Table 1: Experimental results.

| number | model | minDCF | EER(%) |
|---|---|---|---|
| 01 | i-vector | 0.6307 | 13.8665 |
| 02 | x-vector | 0.6027 | 12.1487 |
| 03 | FBANK | 0.5588 | 9.9296 |
| 04 | PCEN | 0.5802 | 10.3858 |
| 05 | WavLM | 0.5884 | 10.6899 |
| 06 | FBANK AttentionConv | 0.6000 | 10.9096 |
| 07 | fusion 1&2 | 0.5754 | 11.6868 |
| 08 | fusion 3&4 | 0.5588 | 9.9296 |
| 09 | fusion 3&5 | 0.4588 | 7.0459 |
| 10 | fusion 1&4 | 0.5558 | 10.0253 |
| 11 | fusion 1&2&3&4&5 | 0.4339 | 6.7920 |

## 4. Resource (Optional)

- i-vector:
github.com/kaldi-asr/kaldi/tree/master/egs/cnceleb

- x-vector:
github.com/kaldi-asr/kaldi/tree/master/egs/cnceleb

- pre-trained WavLM:
github.com/microsoft/unilm/tree/master/wavlm

## 5. References

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

[2] Y. Wang, P. Getreuer, T. Hughes, R. F. Lyon, and R. A. Saurous, "Trainable frontend for robust and far-field keyword spotting," IEEE, 2017.

[3] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, and X. Xiao, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," arXiv e-prints, 2021.

[4] Yue Fan, JW Kang, LT Li, KC Li, HL Chen, ST Cheng, PY Zhang, ZY Zhou, YQ Cai, and Dong Wang, "CN-Celeb: a challenging chinese speaker recognition dataset," in ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020, pp. 7604–7608.

[5] Lantian Li, Ruiqi Liu, Jiawen Kang, Yue Fan, Hao Cui, Yunqi Cai, Ravichander Vipperla, Thomas Fang Zheng, and Dong Wang, "CN-Celeb: multi-genre speaker recognition," Speech Communication, 2022.

[6] J. Kahn, M. Riviere, W. Zheng, E. Kharitonov, and E. Dupoux, "Libri-light: A benchmark for asr with limited or no supervision," in ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020.

[7] G. Chen, S. Chai, G. Wang, J. Du, W. Q. Zhang, C. Weng, D. Su, D. Povey, J. Trmal, and J. Zhang, "Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio," arXiv e-prints, 2021.

[8] C. Wang, M Rivière, A. Lee, A. Wu, and E. Dupoux, "Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation," 2021.

[9] David Snyder, Guoguo Chen, and Daniel Povey, "MUSAN: A Music, Speech, and Noise Corpus," 2015, arXiv:1510.08484v1.

[10] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in IEEE International Conference on Acoustics, 2017.

[11] D. S. Park, W. Chan, Y. Zhang, C. C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," Interspeech 2019, 2019.

[12] L. N. Smith and N. Topin, "Super-convergence: Very fast training of neural networks using large learning rates," 2017.