The speaker verification system description to VoxCeleb Speaker Recognition 2023

Xiaoran Sun[1] Xupu Cai [2]

China Mobile (Hangzhou) Information Technology Co., Ltd.
sunxiaoran@cmhi.chinamobile.com

Abstract

This paper describes team cmhichinamobile submission to track1 and track2 for VoxCeleb Speaker Recognition Challenge 2023(VoxSRC2023). Our best system achieves minDCF0.3265 and EER 4.7950% in track 1, minDCF 0.3059 and EER 4.6420% in track 2.In both Track 1 and Track 2, we used the cam++ model, and in Track 2, we added three Chinese speaker data to obtain a more robust training set.

Index Terms: speaker recogbition, VoxSRC2023, cam++

1. System Description

The VoxSRC-2023 consists of four tracks, the first of which we participated in the first and second tracks, the first track is a closed set of speaker recognition, requiring the participants to use only the Voxceleb2 dev set as a training set, while Track 2 is an open set of speaker recognition, contestants can use any data set other than the test set.In recent years, as more and more researchers focus on deeper and narrower networks in tasks related to speaker recognition, networks of Time delay neural network (TDNN) and its variants, such as ecapa-TDNN, have blossomed in the field of speaker recognition. Recently, the cam++ model was published by Speech Lab, Alibaba Group[1], which has attracted our attention for both accuracy and computational efficiency, and we mainly used this model in VoxSRC2023.

In track1 and track2, cam++ model[1] was used. The cam++ model is a universal model for industrial level speaker recognition officially opened to the public by the Speech Lab, Alibaba Group, which combines accuracy and computational efficiency[1]. In track2, we used an additional non-public Chinese speaker data set with the aim of building a more robust training set. This data set was purchased by our team from Beijing Haitian Ruisheng Science Technology Ltd (SpeechOcean, www.speechocean.com).

2. Data preparation

The voxceleb2 dev dataset was used as our training set in track 1,and it contains 1,092,009 utterances and 5,994 speakers in total. Due to data augmentation making the system more robust, we first use the SoX speed function with speeds 0.9 and 1.1 to generate extra twice speakers[2]. Through the above data augmentation method, we obtained 3276027 discourses and 17982 speakers. Then, we also used RIR[3] and MUSAN[4] to enhance the data, with 20% of the data using RIR data for reverberation enhancement and 20% using MUSAN for noise data enhancement.

In track 2, we used voxceleb 1 dev dataset [5],voxceleb2 dev dataset[6] and the chinese speaker dataset purchased by our team. All data contains 3201120 utterances and 12101 speakers. We used speed enhancement and used 20% of the data for noise data enhancement and 20% for reverberation data enhancement.

The 80-dimensional log-Mel Filter Banks(Fbank) was extracted as input features and without voice activity detection(VAD)[7]. The frame length is 25ms and the frame shift is 10ms. We extracted Fbank based on Torchaudio[8].

3. Model Struction

3.1. FCM

The cam++ model mainly consists of two parts, the front-end convolution module(FCM), and a D-TDNN backbone model[1].The concrete structure is shown in Figure 1.The FCM structure consists of several two-dimensional convolutions with residuals, and finally flatten in the time and frequency domains.This structure can capture more information. In general, TDNN-based models setup more convolution kernels to obtain more detailed frequency information of the input features[9], such as ECAPA-TDNN[10]. The FCM front-end network was used prior to TDNN improves the ability to capture frequency details while reducing network parameters in the TDNN layer[1].

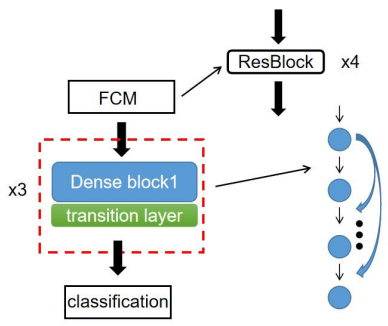


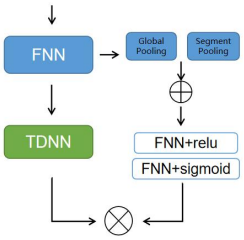
Figure 1: *The basic structure of the model.*

3.2. D-TNN

Each D-TDNN input contains all previous D-TDNN outputs, as shown in formulation1:

$$S^1 = H_1([S^0, S^1, ..., S^{l-1}]) \quad (1)$$

where S^1 is the output of lth D-TDNN layer, and H_l denotes the non-linear transformation. Such a design can enable the network to learn more features, improve the performance of the model[1]. Attention-based context-aware masking (CAM) module was add behind D-TDNN layer for focusing on critical information from noise and features unrelated to the Speaker's information[11]. The segment-level pooling and frame-level pooling have been used as a context mask to learn more speaker information, each output feature of layer has been applied an attention mechanism[1], specifically, the global pooling was used to obtain frame-level context information, while segment pooling was used to obtain segment-level context information, which we regarded as 1 segment per 100 frames according to paper[1].



As illustrated in Figure 2[1], the fbank input is passed through the FCM layer and fed into the TDNN module via a Feedforward neural network (FNN). Simultaneously, the output features of the FNN were globally pooled and segment pooled. These two pooling outputs were then combined using two layers of FNN, with the activation functions of Relu and sigmoid. These to the D-TDNN outputs for calculation. This structure enables the effective extraction of multi-granularity pooling information from the speech input features [1].

3.3. Loss function

The loss function used by our system is additive angular margin loss[12], which has been widely used in many speaker recognition models, such as ECAPA-TDNN[10].

It is based on the concept of angular margin and aims to enhance the discriminative power of features in the feature space[12].Additive Angular Margin Loss introduces additional angular boundaries within the Softmax loss function, which restricts the angles of feature vectors to a predefined range. This clustering of feature vectors within specific angle intervals reduces the angular differences between vectors of the same class, while increasing the angular differences between vectors of different classes[12]. This increased angular margin helps improve the robustness and generalization ability of the model, thereby enhancing the accuracy and reliability of speaker recognition.

3.4. Training Protocol

We used the Open Source cam++ code[1] of Speech Lab, Alibaba Group for training based on the pytorch. The GPU was a Tesla V100 with 16GB of memory. SGD was used for training with the learning rate 0.1, momentum 0.9, and weight 0.0001.

4. Results

4.1. Track1

Table 1 shows the performance of our model in track 1,and without applying score normalization.The results are different from those reported in the paper [1] . The results in the VOXCELEB1-O test set are slightly lower than those in the paper [1] , while the results in VoxCeleb1-E and VoxCeleb1-H are slightly better than those reported in the paper [1] , this may be due to differences in the equipment used for training and configuration parameters.

Table 1: *cam++ performance on VoxCeleb official evaluation sets in track1*

	EER(%)	**MinDCF**
VoxCeleb1-O	0.835	0.0819
VoxCeleb1-E	0.8779	0.1016
VoxCeleb1-H	1.7411	0.1824

In addition to testing on the official VoxCeleb data set, we also tested on the validation set provided by VoxSRC2023.As shown in Table 2, the results on the VOXSRC validation set show an EER of 3.65% and MINDCF of 0.078.

Table 2: *cam++ performance on* VoxSRC2023 *sets in track1*

	EER(%)	**MinDCF**
VoxSRC2023-dev	4.5506	0.2649
VoxSRC2023-test	4.7950	0.3265

4.2. Track2

In Track 2, along with the validation set of Voxceleb 1 versus Voxceleb2, we incorporated a non-public Chinese dataset for training purposes. The outcomes of this approach are depicted in Table3. As our previous research primarily concentrated on Chinese speaker recognition, we further assessed our system in two non-public chinese dataset. EAPA-TDNN have been used to compare with our system, ensuring that the data preparation for ECAPA-TDNN aligned with the training strategy.

Table 3: *cam++ performance on* VoxSRC2023 *sets in track2*

	EER(%)	**MinDCF**
	Cam++	
VoxCeleb1-O	0.617	00614
Chinese-test1	4.6525	0.3946
Chinese-test2	2.385	0.2439
VoxSRC2023-dev	3.66	0.206
VoxSRC2023-test	4.642	0.3059
	ECAPA-TDNN	
VoxCeleb1-O	0.8457	0.0729
Chinese-test1	5.4005	0.3253
Chinese-test2	2.1136	0.1433

This study. Chinese-test 1 consists of 44,947 statements from 300 speakers, while Chinese-test 2 contains 44,922 statements from the same group of speakers. The only difference between the two datasets is the acquisition device used, with Chinese-test 1 on android devices and Chinese-test 2 on Symbian devices. The results indicate that the cam++ model

outperformed the ECAPA-TDNN model in both the VoxCeleb1-O and Chinese-test1 datasets. However, in the Chinese-test2 dataset, the cam++ model had a slightly higher Equal Error Rate (EER) of 2.385% ECAPA-TDNN's EER of 2.1136%.

5. Conclusions

In our approach, we utilized an open-source cam++ model and incorporated a non-public Chinese dataset for track 2. Our system achieved an Equal Error Rate (EER) of 4.7950 and a Minimum Detection Cost Function (minDCF) of 0.3265 on the track. For track 2, our system achieved an EER of 4.6420% and a minDCF of 0.3059.

6. References

[1] H. Wang, S. Zheng, Y. Chen, L. Cheng, and Q. Chen, 'CAM++: A Fast and Efficient Network for Speaker Verification Using Context-Aware Masking', 2023, doi: 10.48550/ARXIV.2303.00332.

[2] H. Yamamoto, K. A. Lee, K. Okabe, and T. Koshinaka, 'Speaker Augmentation and Bandwidth Extension for Deep Speaker Embedding', in Interspeech 2019, 2019.

[3] 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 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2017, pp. 5220–5224.

[4] D. Snyder, G. Chen, and D. Povey, 'MUSAN: A Music, Speech, and Noise Corpus', Comput. Sci., 2015.

[5] A. Nagrani, J. S. Chung, and A. Zisserman, 'Voxceleb: a large-scale speaker identification dataset', ArXiv Prepr. ArXiv170608612, 2017.

[6] J. S. Chung, A. Nagrani, and A. Zisserman, 'VoxCeleb2: Deep Speaker Recognition', in Interspeech 2018, Sep. 2018, pp. 1086–1090. doi: 10.21437/Interspeech.2018-1929.

[7] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, and Y. Bengio, 'SpeechBrain: A General-Purpose Speech Toolkit', 2021.

[8] Y.-Y. Yang et al., 'Torchaudio: Building blocks for audio and speech processing', in ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022, pp. 6982–6986.

[9] W. Wang, D. Cai, X. Qin, and M. Li, 'The DKU-DukeECE Systems for VoxCeleb Speaker Recognition Challenge 2020', 2020.

[10] B. Desplanques, J. Thienpondt, and K. Demuynck, 'ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification', 2020.

[11] Y.-Q. Yu, S. Zheng, H. Suo, Y. Lei, and W.-J. Li, 'Cam: Context-aware masking for robust speaker verification', in ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2021, pp. 6703–6707.

[12] J. Deng, J. Guo, and S. Zafeiriou, 'ArcFace: Additive Angular Margin Loss for Deep Face Recognition', 2018.