System Description for the CN-Celeb Speaker Recognition Challenge 2022

Guangxing Li[1,†], *Wangjin Zhou*[2,†], *Sheng Li*[3], *Yi Zhao*[4], *Hao Huang*[1,*], *Jichen Yang*[5]

[1]School of Information Science and Engineering, Xinjiang University, Urumqi, China
[2]Graduate School of Informatics, Kyoto University, Sakyo-ku, Kyoto, Japan
[3]National Institute of Information and Communications Technology (NICT), Kyoto, Japan
[4]Kuaishou Technology, Beijing, China
[5]School of Cyberspace Security, Guangdong Polytechnic Normal University, Guangzhou, China

ligx2022@gmail.com, zhou.wangjin.54r@st.kyoto-u.ac.jp, sheng.li@nict.go.jp,
zhaoyi07@kuaishou.com, hwanghao@gmail.com, NisonYoung@163.com

Abstract

CN-Celeb is a popular Chinese data set used for speaker recognition. In response to The CN-Celeb Speaker Recognition Challenge 2022 (CNSRC 2022), improving the speaker recognition effect of CN-Celeb, we participated in the task1 of CNSRC 2022, which only used CN-Celeb1 & 2 [1, 2] data sets for training. This paper mainly makes efforts in two aspects: applying scenario information and using the Self-Supervised Learning (SSL) model to the speaker recognition system. Compared to the SSL model, scenario information, named domain embedding, is more effective. We integrate speaker embedding and speaker scenario information abstracted as word embedding and puts forward several back-end processing methods. These methods are more suitable for each registered speaker with multiple enroll utterances. Moreover, data augmentation is also investigated. Experiments show that concatenating and averaging each utterance's domain embedding dimensions of multiple registered speakers with data augmentation shows better performance. Our proposed method effectively reduces 21.3% and 21.5% on EER and minDCF(P=0.01), respectively, compared with the baseline system. Moreover, we did not observe improvement by using SSL as reported in previous works, and the possible reasons and analysis are also given in this paper.

Index Terms: speaker recognition, speaker verification, speaker scenarios, multiple enroll utterances

1. Introduction

The CN-Celeb Speaker Recognition Challenge 2022 (CNSRC 2022)[1] aims to evaluate how well the current speaker recognition methods work in real world scenarios, usually with in-the-wild complexity and real-time processing speed. The challenge is based on CN-Celeb, a large-scale free database with the most real-world complexity so far.

In this CNSRC 2022 challenge, we participated in the fixed track of task 1. This track requires only CN-Celeb-T as the training set, and non-speech data set is allowed for data augmentation. According to the baseline speaker recognition system provided in ASV-Subtools [3], a speaker classifier model is first trained using the CN-Celeb-T, and this model output speakers in the last layer. We use the CN-Celeb-E to extract speaker embedding before the test stage's full connection layer.

We made some improvements to the baseline system.

- Firstly, we observed that, even for speaker recognition, the extracted speaker embeddings could not be directly used as features for testing. The scores from different speaker scenarios of the baseline system are quite different, even from the same speaker. Therefore, it is possible to concatenate speaker scenario information to improve the speaker embedding.
- Besides, to suppress the impact of speaker scenario information on speaker embedding in the same scenario as much as possible, we propose several back-end testing methods.
- Moreover, in order to make the training data more diverse, we also investigate the effectiveness of data augmentation using the non-speech datasets.
- Finally, we also tried to introduce Self-Supervised Learning (SSL) models to extract speaker embeddings, inspired by the methods of the top teams [4, 5] in the VoiceMOS Challenge 2022.

In the rest of this paper, we will briefly introduce the system we submitted on CNSRC 2022. We will describe the method we proposed in section 2 and describe the specific process of the experiment in section 3, including the back-end test method we proposed. In addition, we will summarize the experimental results in Section 4.

2. Proposed Method

In the first task of the CNSRC 2022 provided in ASV-Subtools [3], the CN-Celeb has many scenarios, such as speech, singing, and interview. We referred to the different scenarios of CN-Celeb as domains, and there are 11 different domains in the CN-Celeb data set. Based on this, we tried to abstract the domain labels of each utterance into different word embeddings in jointly training SE-ResNet34 [6]. We concatenate the word embedding and speaker embedding and name the concatenated embedding as domain embedding. The benefit of domain embedding for the speaker recognition task was found in subsequent tests.

Another feature of the CN-Celeb data set compared to other data sets is that each speaker can enroll multiple utterances. However, the general processing method is to concatenate multiple utterances of a speaker into one utterance. And then, the concatenated utterance is used as the enrollment. We used various testing methods when testing the effect of domain embedding on speaker recognition. We found that enrolling multiple

[†]Equal contribution. * Corresponding Author.
[1]http://www.odyssey2022.org/col.jsp?id=102

utterances for a single speaker significantly influences recognition. Moreover, we verified that the universality of enrolling multiple utterances for a single speaker could effectively improve the baseline system's recognition.

More detail speaking, we use the learnability of word embedding to abstract these 11 domains into different 32-dimensional embeddings. Then we concatenate them after the 256-dimensional embedding calculated by SE-ResNet34 [6] during training. We first try to maximize the same-speaker similarity and minimize the different-speaker similarity scores in the testing phase. The specific method is to find the rules by analyzing the cosine scores of test utterances and the corresponding registered utterances. We want to filter out some test-enrollment pairs with high scores as positives and others as negatives.

Besides, in order to make the training data more diverse, we also investigate the effectiveness of data augmentation using the non-speech datasets.

SSL can capture rich features because it is trained in large-scale data sets. There are some precedents that using SSL for speaker recognition, [7] fine tune in wav2vec 2.0 [8,9] based on Vox-Celeb [10, 11] data set, [12] fine tune in wav2vec 2.0 [8, 9] based on NIST SRE [13,14] series data sets, Vox-Celeb [10,11] and several Russian data sets, and [15] has a number of state-of-the-art results in SUPERB, which has surprising results in speaker recognition. However, none of them is fine tuned on the Chinese dataset, so we want to verify the effect of various pre-training models on the speaker recognition task on the Chinese dataset. We investigate using SSL models to extract speaker embeddings, inspired by the methods of the top teams [4, 5] in the VoiceMOS Challenge 2022.

3. Experimental Settings

3.1. Data

All experiments are conducted on the CN-Celeb data sets.

- **Training set**: The CN-Celeb1 & 2 [1,2] train sets, which contain more than 600,000 audio files from 3000 speakers, are used for training.
- **Enrollment set**: The enrollment set is from CN-Celeb1, which contain 799 audio files from 196 speakers.
- **Testing set**: The testing set is from CN-Celeb1, which 17777 audio files from 200 speakers.

Compared with enrollment speakers, four speakers in the test set are out-of-set speakers. Note that both the enrollment and test sets do not intersect with the training set. In the experiments, we focus on speaker recording scenarios. The CN-Celeb datasets annotate the recording scenarios for each speech, which provides an essential guarantee for our experiments. There are the 11 domains in CN-Celeb, and the 11 domains in CN-Celeb is shown in Table 1.

3.2. Baseline system

Our baseline system refers to the implementation of CNSRC 2022 task 1 provided by ASV-Subtools [3]. For the input features, 81-dimensional filter banks are extracted within a 25ms sliding window for every 10ms, and then we used Voice Activity Detection(VAD) to remove silence frames. SE-ResNet34 [6] is used in baseline system to train a classification network, which AM-Softmax [16] is used as loss function. And then we extract 256-dimensional speaker embedding for enrollment dataset and test dataset respectively after training.

Table 1: Distribution of 11 scenarios in CN-Celeb 1 and 2

domains	#Spks	#Utters	#Hours
Entertainment	1099	54046	94.51
Interview	1299	93341	217.05
Singing	712	54708	104.12
Play	196	19237	26.99
Movie	195	7198	7.97
Vlog	529	126187	181.15
Live Broadcast	617	175766	456.30
Speech	516	35081	118.80
Drama	428	20363	22.75
Recitation	259	60978	134.16
Advertisement	83	1662	4.04
Overall	3000	659594	1363.74

In addition, enroll utterances belonging to the same speaker are concatenated into one utterance for enrollment, and all of our experiments are based on cosine similarity as the back-end scoring criterion.

3.3. Domain embedding

It is a major feature to introduce the scenario information of the utterances in our system. The speaker's scenario information is absolutely non-negligible. The voiceprint characteristics of the same speaker in the case of speech and singing are quite different, which is also confirmed by the scoring results in the baseline system. In considering this, our goal is to introduce the speaker scenarios information and make the model learn the scenario features in the training process. In the test process, we hope to make the similarity score of speakers in the same scenario as their score as far as possible. We can reduce the recognition errors caused by the differences in speaker scenarios in this way.

The specific implementation method is as follows. Input information of the model includes 81-dimensional FBank acoustic features and scenario information corresponding to the input utterance. With the preprocessing of VAD, 256-dimensional speaker embedding is calculated through the input acoustic features. Then, the corresponding 32-dimensional word embedding is calculated through the incoming speaker scenario information. Finally, the 256-dimensional speaker embedding and 32-dimensional word embedding are concatenated in series to form a 288-dimensional domain embedding, which we can extract embeddings for the speakers. The model structure is shown in Figure 1.

Table 2: Multi enroll for Baseline systems and Domain Embedding. Base1 and Base2 respectively indicate that non-speech data sets are not used and used for data augmentation, corresponding to System1 and System2.

	Baselines		Domain Embedding	
	Base1	Base2	System1	System2
EER(%)	17.611	15.067	16.542	14.784
minDCF(P=0.01)	0.7335	0.6996	0.7129	0.6785

3.4. Different back-ends

With the scenario information is introduced into the training, the back-end test method will be different from the baseline system.

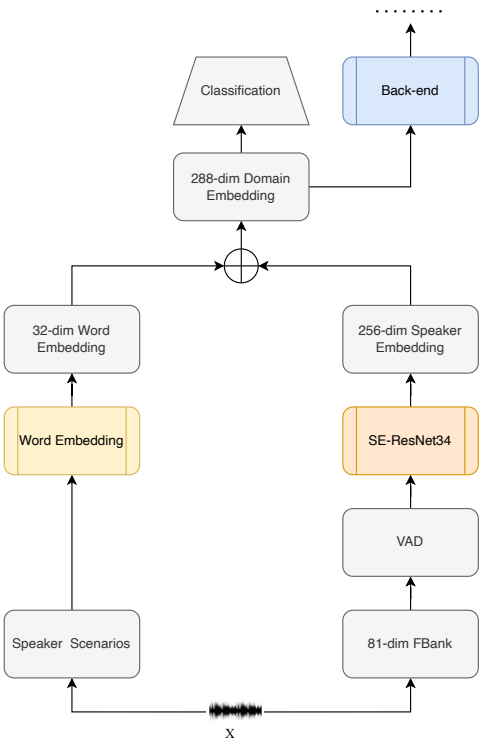

Figure 1: The structure of Domain Embedding. x is the input utterance, which extracted the information of speaker scenarios and acoustic features, respectively. The Back-end approaches are showed in Subsection 3.4.

Because the baseline system concatenates multiple enrollment utterances into single utterance as the enrollment, the registered voice loses the speaker scenario information, which is inconsistent with our test settings. In order to retain the scenario information of enrollment, we will register multiple utterances of the speakers in the enrollment set, and each test utterance will make cosine similarity score with 799 utterances of 196 enroll speakers during the testing. However, the results showed that there is no significant improvement, because the test method is not the comparison of the same scenario. We propose the following four solutions to solve this problem.

- **Max_All**: We want to obtain the maximum score of each test utterance to the utterance of the enroll speakers as the score of this speaker and each enroll speaker when testing.
- **Max_1_Min**: Based on the first method, we take the maximum score of the test utterance and the enroll utterance as the score of the speaker corresponding to the test utterance and the enrollment utterance, and select the minimum score of the test utterance and other utterance in the enrollment set as their similarity score.
- **Max_10_Min** and **Max_20_Min**: Through a simple example analysis of the score histogram of the test utterances and the enrollment utterances, which shown in Figure 2, we maximize the scores of 10 pairs and 20 pairs as the scores of the test speech and the enrollment speech respectively, and minimize the other score pairs as the scores of the test set speaker corresponding to each registered speaker.
- **Concate_Ave**: For multiple enroll utterance of the speaker in the enrollment set, we concatenate the speaker embeddings of multiple utterances by dimensions, and then average it as the enroll speaker embedding, which is the approach we submit in the challenge.

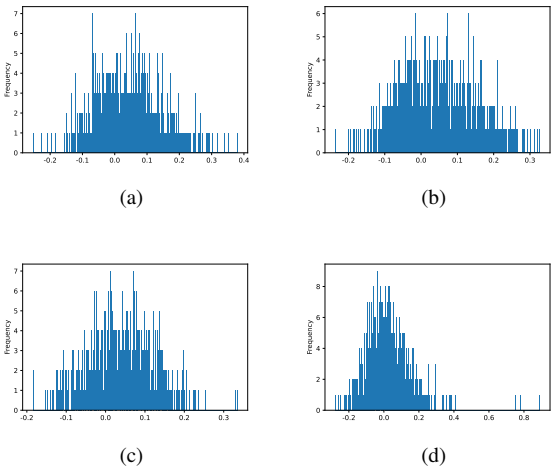

Figure 2: Score histogram of the test utterances and the enrollment utterances. a, b, c and d are the score histogram examples of 4 test utterances to enrollment utterances. The horizontal axis is the score, and the vertical axis is the number of enrollment utterances corresponding to the score.

3.5. Data augmentation

In order to make the training data more diverse, we used the non-speech datasets (MUSAN [17] and RIRS NOISE [18]) for data augmentation, which finally doubled the data size.

3.6. Self-Supervised Learning

In order to obtain advanced effects, we spent a lot of time for SSL, mainly using four pre-training models in the experiment, including wav2vec_small, wav2vec_vox_new, hubert_large [19] and wavLM_base, using CN-Celeb-T's random 100 hours and full volume data for fine tuning respectively. In these fine-tuning processes, we used the same loss function as the baseline system, AM-Softmax [16], to obtain better classification results, and all of them trained with the fairseq [20].

Because SSL models can extract robust speaker embedding, we do not design complex downstream tasks but add a statistical pooling layer and some linear layers to obtain fixed dimension speaker embedding. However, the experimental results are not as expected, and the specific analysis is shown in Subsection 4.2.

4. Experimental Evaluations

4.1. Experimental Results

Our experimental results are shown in Table 2 and Table 3, which are mainly divided into two parts: baseline system and joint training based on baseline system with speaker scenarios information.

Table 3: EER and minDCF(P=0.01) of baselines and our systems with different back-ends. There are mainly two systems which include Baseline systems and joint training with domain embedding, which use different back-end approaches mentioned in section back-end. Y means that it uses non-speech datasets, i.e., MUSAN [17], and RIRS NOISE [18] for data augmentation, and N means without augmentation.

	Baselines		Domain Embedding + Different back-end									
	Base1	Base2	Max_All		Max_1_Min		Max_10_Min		Max_20_Min		Concate_Ave	
	N	Y	N	Y	N	Y	N	Y	N	Y	N	Y
EER(%)	11.450	9.580	11.000	10.270	11.631	10.696	9.952	9.378	9.655	9.085	9.784	**9.006**
minDCF(P=0.01)	0.5357	0.4919	0.4711	0.4398	0.4273	0.3847	0.4711	0.4397	0.4711	0.4398	0.4765	**0.4248**

- The baselines include two systems. One totally follows the baseline training of ASV-Subtools [3], and the other is the Base2 system, which uses non-speech datasets, MUSAN [17], and RIRS NOISE [18] for data augmentation compared with the Base1 system.
- Domain embedding is the system that we proposed, and the different back-end approaches mentioned in Subsection 3.4 are used for comparison. In addition, the N and Y are the abbreviations of No and Yes, which stands for with and without using data augmentation in experiments, respectively.

By comparing the EER and minDCF of the baseline systems and the domain embedding systems in Table 2, when a speaker enrolls multiple utterances, we can find that adding the speaker scenarios information in the training process reduces both the EER and minDCF significantly.

Besides, we can find the following conclusions through the comparisons in Table 3. Firstly, whether comparing the Base1 system with columns "N" of the domain embedding system or comparing the Base2 system with columns "Y" of the domain embedding system, we can find that these back-end approaches significantly improve the effectiveness of the task. The last three back-end testing methods of domain embedding, Max_10_Min, Max_20_Min, and Concat_Ave, all significantly affect EER and minDCF. After synthesizing the two measurement indicators of the speaker recognition task, we submitted the Concat_Ave as our final result.

4.2. Further Discussions

It is worth mentioning that with the rise of SSL in speech recognition, we hope that using SSL can also improve the speaker recognition task. Both [7, 12] use wav2vec 2.0 fine-tuned with their data sets for English-speaking speaker recognition. Moreover, [7] almost entirely follows the wav2vec 2.0 model structure to realize speaker recognition and language recognition, while [12] uses wav2vec 2.0 to extract better features and access TDNN [21] twice to filter out better speaker embeddings. The experiments of both methods show that the first several transformer layers can better classify speakers. We had hoped to fine-tune multiple models on the CN-Celeb data set and fuse the fine-tuning results to improve the effect. Unfortunately, the fine-tuning results of these four models are not satisfactory, and the EER of the best result is still greater than 14%. Based on this situation, we have the following three assumptions:

- **Different languages**: The pre-training process of the above models are based on English, which the fine-tuning effect on the Chinese data set is poor.
- **Simple classification network**: The classification network is too sample to achieve the desired effect after extracting speaker features through SSL.
- **Complex model**: It can be seen from [7, 12] that the overly complex model plays an excessive role in the task of speaker recognition.

Based on the above analysis, the most likely reason for less-perfect classification results might be that SSL is overly strong in capturing too much unnecessary information, but not the essential voiceprint features for speaker recognition. We will focus on improving it in future research.

In addition, although we did not mention it in Section 3, CN-Celeb does have a long tail problem in each speaker scenario. The unbalanced data distribution must lead to biased results inconsistent with our expectations in the training process. Therefore, we can use resampling to obtain a more balanced data set for training in the subsequent experiments.

5. Conclusion

In this paper, to effectively test under different scenarios, we propose domain embedding, the speaker scenario word embedding combined with the speaker embedding. From the experiments, the proposed domain embedding outperforms the conventional speaker embedding. In addition, to test in the same scenario as much as possible, we investigate several back-ends with data augmentation strategies and select the most effective one. Although the SSL model did not show its superiority in our experiment, it should be a promising direction in the future, and we will continue to work on it. Another observation is that it is very challenging to use a general speaker recognition system to recognize some specific scenarios, such as singing and drama, the voiceprint of which is unclear. Therefore, the data processing method has much room for improvement in the future.

6. References

[1] Y. Fan, J.W. Kang, L.T. Li, K.C. Li, H.L. Chen, S.T. Cheng, P.Y. Zhang, Z.Y. Zhou, Y.Q. Cai, and D. Wang, "Cn-celeb: A challenging chinese speaker recognition dataset," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7604–7608.

[2] 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.

[3] Fuchuan Tong, Miao Zhao, Jianfeng Zhou, Hao Lu, Zheng Li, Lin Li, and Qingyang Hong, "Asv-subtools: Open source toolkit for automatic speaker verification," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6184–6188.

[4] Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari, "UTMOS: UTokyo-SaruLab System for VoiceMOS Challenge 2022," *arXiv e-prints*, p. arXiv:2204.02152, Apr. 2022.

[5] Zhengdong Yang, Wangjin Zhou, Chenhui Chu, Sheng Li, Raj Dabre, Raphael Rubino, and Yi Zhao, "Fusion of self-supervised learned models for mos prediction," *arXiv e-prints*, pp. arXiv–2204, 2022.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[7] Zhiyun Fan, Meng Li, Shiyu Zhou, and Bo Xu, "Exploring wav2vec 2.0 on speaker verification and language identification," *arXiv preprint arXiv:2012.06185*, 2020.

[8] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli, "wav2vec: Unsupervised pre-training for speech recognition," *arXiv preprint arXiv:1904.05862*, 2019.

[9] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12449–12460, 2020.

[10] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[11] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *Interspeech 2018*, 2018.

[12] Sergey Novoselov, Galina Lavrentyeva, Anastasia Avdeeva, Vladimir Volokhov, and Aleksei Gusev, "Robust speaker recognition with transformers using wav2vec 2.0," *arXiv preprint arXiv:2203.15095*, 2022.

[13] Douglas Reynolds, Elliot Singer, Seyed O Sadjadi, Timothee Kheyrkhah, Audrey Tong, Craig Greenberg, Lisa Mason, and Jaime Hernandez-Cordero, "The 2016 nist speaker recognition evaluation," Tech. Rep., MIT Lincoln Laboratory Lexington United States, 2017.

[14] Seyed Omid Sadjadi, Craig Greenberg, Elliot Singer, Douglas Reynolds, and Jaime Hernandez-Cordero, "The 2018 nist speaker recognition evaluation," in *Interspeech 2019*, 2019.

[15] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, et al., "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *arXiv preprint arXiv:2110.13900*, 2021.

[16] Feng Wang, Jian Cheng, Weiyang Liu, and Haijun Liu, "Additive Margin Softmax for Face Verification," *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 926–930, July 2018.

[17] David Snyder, Guoguo Chen, and Daniel Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[18] Tom Ko, Vijayaditya Peddinti, Daniel Povey, Michael L Seltzer, and Sanjeev 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.

[19] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[20] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli, "fairseq: A fast, extensible toolkit for sequence modeling," *arXiv preprint arXiv:1904.01038*, 2019.

[21] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *Interspeech 2017*, 2017.