



The Sogou System for Short-duration Speaker Verification Challenge 2021

Jie Yan, Shengyu Yao, Yiqian Pan, Wei Chen

Sogou-Inc. , Beijing, China

{jieyan, yaoshengyu, panyiqian, chenweibj8871}@sogou-inc.com

Abstract

In this paper we present our system for the task 2 of the Short-duration Speaker Verification (SdSV) Challenge 2021. This task focuses on benchmarking and varying degrees of phonetic variability analysis of short-duration speaker recognition system. The main difficulty exists in the variance between cross-lingual trials, along with the limited in-domain Farsi training data. Based on the state-of-the-art ResNetSE speaker embedding network, we propose a novel network architecture with in-domain data finetuning and novel scoring methods, and achieve significant improvement over the ResNetSE baselines. Furthermore, score calibration on duration efficiently improve the robustness. Finally, our system with fusion of 10 subsystems achieve satisfying results in MinDCF and EER of 0.0394 and 0.84% respectively on the SdSVC evaluation set.

Index Terms: speaker recognition, ResNet models, SdSV challenge

1. Introduction

Speaker verification refers to the task of determining whether a test audio was spoken by a target speaker. The Short-duration Speaker Verification (SdSV) Challenge 2021 focuses more on the short-duration and phonetic variability problems in speaker verification. This task is designed with a fixed training dataset consisting of VoxCeleb1 [1], VoxCeleb2 [2], LibriSpeech [3] and a part of the DeepMine corpus [4]. Trials consists of Farsi enrollment utterances and a test utterance which can be either Farsi or English speech. This brings the problem that most of the training data except DeepMine are spoken in English, making it hard to get satisfying performance on out-domain data spoken in Farsi. For this reason, the key issue is to build robust speaker verification system to reduce the language mismatch with the constraint of audio duration.

We solve this problem in three ways. Firstly, we improve the structure of baseline ResNetSE models, change the sequence of layers in residual block to train and generalize better. Moreover, we investigate Angular prototypical Loss [5] and circle loss [6] in our work. These modifications bring obvious improvement in speaker embedding extractor. Secondly, after the speaker embeddings are extracted, several novel scoring methods based on cosine similarity and PLDA are used for scoring. To reduce the mismatch in duration of enrollment and test utterance, we take an enhanced cosine score and reverse score procedure. Finally, system fusion is used to obtain the final submission by bosaris toolkit [7].

The rest of the paper is organized as follows. In section 2, we introduce the dataset and our data preparation for this challenge. Section 3 describes our Sogou submission, including the model architectures and back-ends. Section 4 shows the detail performance and the analysis. Finally, Section 5 concludes the paper with final remarks.

2. DataSet

2.1. Training data

According to the evaluation plan of SdSV 2021, the task 2 is a fixed training condition where the system should only be trained using a designated set. The available training data is limited as follow:

- VoxCeleb1
- VoxCeleb2
- LibriSpeech
- Mozilla Common Voice Farsi
- DeepMine (Task 2 Train Partition)

Among these datasets, VoxCeleb 1, Voxceleb 2 and LibriSpeech contain only English speakers which has language mismatch to the evaluation set. DeepMine data for task 2 only contains text-independent Persian utterances from 588 speakers. Mozilla Common Voice are also Farsi speakers but no exact speaker labels were supplied. It is forbidden to use any other public or private speech data for training.

2.2. Test data

In task 2, each trial in the evaluation set contains a test utterance and a target model. The enrollment data for target model consists of one to several variable-length utterances with a duration of roughly 4 to 180 seconds. The duration of the test utterances varies between 1 to 8 seconds. More details can be found in [8].

2.3. Data preparation

In our experiments, we make use of all allowed training data with speaker label except the VoxCeleb1 test partition as our training data, totally 10249 labeled speakers. It is also used for training back-end models such as PLDA. During network training, additive noise from MUSAN corpus [9] and room impulse response (RIR) simulation [10] are used as data augmentation, which is randomly selected in every training step. VAD is not contained in the training phase because there exist very few silence in training speech. But in the test phase, LSTM-VAD is used. To reduce the language mismatch between training data and test data, we finetune the model on the DeepMine corpus which contains only 588 speakers.

3. Sogou submission

This section describes the Sogou SdSVC final submission. Compared with a regular ResNetSE baseline, we propose our improvement on the network and scoring in the subsequent sections.

3.1. Speaker Models

3.1.1. Baseline ResNetSE Model

Residual networks has been widely used in image recognition and recently has been applied to speaker recognition successfully. We use ResNet with Squeeze-and-Excitation(SE) layer as our first speaker model. The network structure can be regarded as three parts: a front extractor which learns a frame-level representation from the input acoustic feature, an encoder layer which aggregates frame-level features into utterance-level representation and produces a fixed dimensional speaker embedding, and a classifier which measures the training loss.

Table 1 shows the detail of the ResNetSE architecture. The input features are first processed by the initial convolution and the following 4 residual blocks. The detail of a residual block with SE module is shown in Figure 1. Then the feature maps are fed into the pooling layer to aggregate frame level information and finally transformed into a fixed dimensional vector for loss functions. In our implementation, Self-attentive pooling (SAP) is used as the encoder layer. The speaker embeddings are 512-dimensional and the loss is the Angular Prototypical loss. We use Adam gradient descent with initial learning rate equal to 0.001 to train the models and the learning rate reduced by 25% every three epochs.

Table 1: *Trunk architecture for ResNetSE model. L : length of the input sequence.*

Layer	Kernel size	Stride	Output
Conv1	$3 \times 3 \times 32$	1×1	$L \times 64 \times 32$
Block1	$\begin{bmatrix} 3 \times 3 \times 32 \\ 3 \times 3 \times 32 \end{bmatrix} \times 6$	1×1	$L \times 64 \times 32$
Block2	$\begin{bmatrix} 3 \times 3 \times 64 \\ 3 \times 3 \times 64 \end{bmatrix} \times 16$	2×2	$L/2 \times 64 \times 32$
Block3	$\begin{bmatrix} 3 \times 3 \times 128 \\ 3 \times 3 \times 128 \end{bmatrix} \times 24$	2×2	$L/4 \times 64 \times 32$
Block4	$\begin{bmatrix} 3 \times 3 \times 256 \\ 3 \times 3 \times 256 \end{bmatrix} \times 3$	2×2	$L/8 \times 64 \times 32$
Flatten	-	-	$L/8 \times 2048$
Pooling	-	-	4096
Linear	512	-	512

3.1.2. ResNetSEV2 Model

As it described in [11], the residual block in ResNet baseline model performs the following computation:

$$y_l = h(x_l) + F(x_l, W_l) \quad (1)$$

$$x_{l+1} = f(y_l) \quad (2)$$

where the F denotes the residual function, x_l is the input feature to the l -th Residual block, the function f is the operation after element-wise addition. In resnet structure, if function h and f are identity mapping, we can get

$$x_L = x_l + \sum_{i=l}^{L-1} F(x_i, W_i) \quad (3)$$

for any layer L recursively. From the chain rule of back-propagation we have

$$\begin{aligned} \frac{\partial loss}{\partial x_l} &= \frac{\partial loss}{\partial x_L} \frac{\partial x_L}{\partial x_l} \\ &= \frac{\partial loss}{\partial x_L} \left(1 + \frac{\partial}{\partial x_L} \sum_{i=1}^{L-1} F(x_i, W_i) \right) \end{aligned} \quad (4)$$

This suggests that if we make f and h an identity mapping the signal can be directly propagated from any unit to another, not only within a residual unit, but through the entire network both forward and backward, which is better for network training.

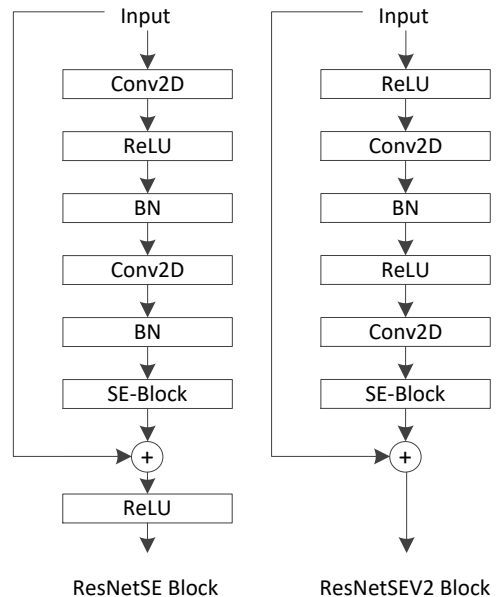


Figure 1: *residual block in our models.*

This can be done by rearranging the activation functions in network. Figure 1 shows the architecture of residual block in ResNetSE and ResNetSEV2 model. In the ResNetSEV2 block, we change the order of network layers by putting the Batch Norm layer (BN) and ReLU function ahead of the convolutions. No extra operations are performed after the element-wise addition. Thus we can create a direct path across residual blocks. Experiments suggest that keeping a direct and simple information path is helpful for optimization. Besides, it improves regularization of the models by regarding BN layer as pre-activation. Based on this unit, we present ResNetSEV2 model, which is much easier to train and generalizes better than the baseline ResNetSE model in 3.1.1.

3.1.3. ECAPA-TDNN model

The ECAPA-TDNN model proposed in [12] is also an outstanding speaker extractor which is based on the original x-vector architecture and put more emphasis on channel attention, propagation and aggregation. We use this model as one of the subsystem in our submission.

3.1.4. Loss Function

As we all know, the loss function is very essential to network training in speaker recognition. AAM-softmax loss functions have been proposed in face recognition and successfully applied to speaker recognition. Angular Prototypical (AP) loss is

Table 2: The EER(%) and MinDCF(*100) results of our subsystems on the Development set and Evaluation set of SdSV Challenge 2021 task 2. 'gender' means finetune with female/male data respectively. 'SC' means score calibration operation.

System	Model	Loss	Finetune	Scoring Method	SC	SdSV Dev		SdSV Eval	
						EER	MinDCF	EER	MinDCF
1	ResNetSE	AP	no	Cosine	yes	1.71	6.62	1.28	6.18
2	ResNetSE	AP	yes	Cosine	yes	1.42	6.29	1.15	5.35
3	ResNetSEV2	AP	no	Cosine	yes	1.32	6.77	1.18	5.52
4	ResNetSEV2	AP	yes	Cosine	yes	1.12	6.04	1.05	4.84
5	ResNetSEV2	CircleLoss	no	Cosine	yes	1.75	8.20	1.52	7.43
6	ECAPA-TDNN	AP	yes	Cosine	yes	1.73	7.51	1.36	6.69
7	ResNetSE	AP	yes	Reverse	yes	1.33	5.55	1.07	4.95
8	ResNetSE	AP	gender	Cosine	yes	1.43	6.29	1.16	5.57
9	ResNetSE	AP	no	PLDA	no	1.70	7.76	1.51	7.08
10	ResNetSEV2	AP	no	PLDA	no	1.29	6.48	1.49	7.18
fusion 1	2+4	-	-	-	-	1.18	5.94	1.01	4.59
fusion 2	2+4+7	-	-	-	-	1.13	4.91	0.95	4.34
fusion 3	1~10	-	-	-	no	1.01	4.48	0.89	4.21
fusion 4	1~10	-	-	-	yes	0.97	3.91	0.84	3.94

a variant of the prototypical networks with an angular objective which has strong performance. Triplet loss is very convenient for model finetune. In our work, we made use of various objective functions in speaker extractor training and some of them works well in fusion system. The objective functions used in our systems are as follows:

- Standard Softmax
- Triplet loss [13]
- AAM-Softmax [14]
- Angular Prototypical Loss [5]
- Circle Loss [6]

3.2. Model Finetune

To reduce phonetic variability caused by language mismatch, we further fine-tune the models with Farsi data to improve the performance on testset. In practice, we take this operation in three ways:

- Finetune the trained model with hard triplet loss.
- Fix the parameter of the speaker network except loss functions and retrain the model with original loss.
- Divide the DeepMine data into two groups by gender and retrain the model with female or male data respectively.

Experiments show the last two methods help extract robust speaker embedding on SdSV 2021 development and evaluation set.

3.3. Scoring

The trained networks are evaluated on the SdSV 2021 development and evaluation sets. We compared different scoring methods on the trained models.

3.3.1. Cosine score

To build a robust speaker verification system, we use an enhanced cosine similarity method to measure whether the two utterances are from the same speaker after the embeddings are extracted. The steps are as follows:

1. For each utterance in the enrollment model, we sample ten 4-second temporal segments, calculate the mean of all these embeddings as the enrollment embedding.

2. For the test utterance, we sample ten 2-second temporal segments, and compute ten cosine similarities in total with the enrollment embedding.
3. Take the mean of these 10 similarities as the final score.

3.3.2. PLDA score

PLDA [15] is a strong back-end model in speaker verification task, especially in the traditional i-vector and x-vector system. To further reduce the language variabilities between training data and testing data, we also use PLDA for scoring. Firstly, whitening is applied with development data. Then, PLDA models are trained on labeled speakers after the speaker embeddings are extracted. In PLDA procedure, we calculate the mean of all the temporal segments embeddings as final embedding both enrollment and test utterance.

3.3.3. Reverse score

The enrollment in SdSV2021 evaluation trials consist of several utterances, while the test utterances are relatively short. Besides, we have different operations on the enrollment and test utterances in our cosine scoring method. By exchanging the positions of enroll and test utterances, we can get more random sampling from enroll and test utterances and focus more on the speaker characteristics.

3.4. Score calibration

The speaker similarity score is largely affected by the quality of the trial speeches [16], hence the score calibration is also applied for cosine similarities scores. Note that, the enrollment data for each model is sufficient compared to the test speech in this task, so we design the quality function mainly based on the duration of the test speech. Inspired by the time penalty in [17], assuming speech duration for enrollment is long enough, the re-scoring method is as follows:

$$S' = S + C \cdot f(d_t) \quad (5)$$

where S is the cosine similarities score, C is a scaling constant, d_t is the duration (in seconds) of test speech in scoring trial, f is the duration based quality function:

$$f(d_t) = \frac{1}{d_t} \quad (6)$$

The optimal value of C is 0.05 in our final fusion systems.

Table 3: The EER(%) and MinDCF(*100) results of PLDA on the Development set with different training data

Data Set	Spk Num.	ResNetSE		ResNetSEV2	
		EER	MinDCF	EER	MinDCF
DeepMine Task2 train	588	1.91	8.28	1.98	8.84
All training set + part of CommonVoice	13234	1.70	7.76	1.29	6.48
SdSV2021 enrollment (Toy experiment)	15555	0.82	4.29	0.61	3.40

4. Experiments and results

4.1. Performance metric

The main performance metric for the challenge is normalized minimum Detection Cost Function (MinDCF) which is defined as a weighted sum of miss and false alarm error probabilities. Moreover, the equal error rate (EER) is also a very important performance metric in speaker verification. In our experiments, we test the MinDCF and EER performance for every temporary model after each epoch on development set and choose the best one.

4.2. System Results

For this submission we train all these models with our training data and data augmentation as described in section 2.3. To reduce the duration mismatch of the training and test data, we randomly select examples with 2 seconds for network training from experience. All the models are trained with 64 dimensional MelSpectrogram features extracted using torchaudio toolkit [18]. Adam gradient descent is used and the initial learning rate equals to 0.001. The learning rate is reduced by 25% every three epochs. The batchsize is set to 200. The DeepMine data is used for model finetune with AP and semi-hard triplet loss.

Table 2 illustrates the results of different speaker embedding extractor on the development and evaluation set. ResNetSEV2 achieves the best performance both in EER and MinDCF by applying all these strategies, as it shows in system 4. The best subsystem results in MinDCF and EER of 0.0484 and 1.05% respectively and the final performance of fusion submission is 0.0394 and 0.84% on the SdSVC evaluation set. Comparing the results of system 1 to system 6, we can find that different model and loss are complementary. No score calibration is applied to all the subsystems in fusion system 3 but only no use to PLDA subsystems in fusion system 4. The results indicate that score calibration improves the MinDCF by 6.5% relatively.

System 5 shows the performance of ResNetSEV2 with circle loss and system 6 is a conventional ECAPA-TDNN model. The MinDCF and EER of system 5 and system 6 are slightly worse compared to system 3, but they all contribute to the fusion system.

System 7 shows the fusion results of the reverse score strategy with ResNetSE model. The MinDCF of this strategy is 0.0555 on SdSV 2021 development set, improved by 12% compared to the traditional cosine result of system 2. Besides, reverse score strategy works well in the fusion systems, as fusion system 2 in table 2 shows.

Considering that female appears more in test but less in training data, we try to fit the model to female data. In system 8, we divide the finetune data into two groups by gender, and then finetune the model with specific data respectively. The result shows this method improves the performance especially in female trials.

Table 4: The EER(%) and MinDCF(*100) results of PLDA on the Development set with different LDA dimension.

LDA dim.	ResNetSE		ResNetSEV2	
	EER	MinDCF	EER	MinDCF
400	1.98	8.14	1.23	6.70
300	1.77	7.76	1.29	6.48
200	1.70	7.76	1.50	7.16
100	1.77	8.00	1.57	6.63

4.3. Compare of ResNetSE and ResNetSEV2

The performance of ResNetSE model (system 1) and ResNetSEV2 model (system 3) described in section 3.1 show in table 2 respectively. It can be found that ResNetSEV2 performs better under the same training strategy, improved by 15%/7% in EER/MinDCF compared with the former. After applying model finetune and score calibration, the ResNetSEV2 achieves the best performance both in EER and MinDCF. Further experiments indicate that PLDA also works well with ResNetSEV2 embeddings. System 9 and 10 show the PLDA score results of these two models. The PLDA improves the performance of ResNetSEV2 model in this mismatch condition but very little improve with ResNetSE model. Besides, the ResNetSEV2 model is insensitive to the training data of PLDA models as table 3 shows. We can see that the performance model have obviously improvement as the number of training speaker increased even though the training data is cross-lingual. If we use SdSV2021 enrollment data of evaluation set for PLDA training, that is, no language mismatch in this situation, the advantage of ResNetSEV2 model is more prominent. Table 4 illustrates the influence of LDA dimension when PLDA model is trained with all training set, and both of the two model achieve best MinDCF with LDA dimension equals to 300.

5. Conclusions

This paper presents the system submitted by Sogou in task 2 of SdSV challenge 2021. To obtain robust systems, we have made various attempts in data processing, network training and scoring. Together with the use of in-domain data finetune and score calibration, the proposed ResNetSEV2 model achieved significant improvement over baselines. The final submission yielded minDCF of 0.0394 and EER of 0.84 on the evaluation subset, which was the 3rd and 4th best result in EER and minDCF of the task.

6. Acknowledgements

The authors gratefully acknowledge the financial support provided by the National Key Research and Development Program of China (2020AAA0108004).

7. References

- [1] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.
- [2] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.
- [3] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.
- [4] H. Zeinali, L. Burget, and J. H. Černocký, "A multi purpose and large scale speech corpus in persian and english for speaker and speech recognition: the deepmine database," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 397–402.
- [5] J. S. Chung, J. Huh, S. Mun, M. Lee, and I. Han, "In defence of metric learning for speaker recognition," 2020.
- [6] Y. Sun, C. Cheng, Y. Zhang, C. Zhang, L. Zheng, Z. Wang, and Y. Wei, "Circle loss: A unified perspective of pair similarity optimization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6398–6407.
- [7] N. Brummer and E. de Villiers, "The bosaris toolkit: Theory, algorithms and code for surviving the new dcf," 2013.
- [8] H. Zeinali, K. A. Lee, J. Alam, and L. Burget, "Short-duration speaker verification (sds) challenge 2021: the challenge evaluation plan," 2021.
- [9] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *Computer Science*, 2015.
- [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] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European conference on computer vision*. Springer, 2016, pp. 630–645.
- [12] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," in *Interspeech 2020*, 2020.
- [13] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.
- [14] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4690–4699.
- [15] S. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *IEEE 11th International Conference on Computer Vision, ICCV 2007, Rio de Janeiro, Brazil, 2007*, 2007.
- [16] J. Thienpondt, B. Desplanques, and K. Demuynck, "The idlab voxsrc-20 submission: Large margin fine-tuning and quality-aware score calibration in dnn based speaker verification," *arXiv preprint arXiv:2010.11255*, 2020.
- [17] N. Torgashov, "Id r&d system description to voxceleb speaker recognition challenge 2020," 2020.
- [18] torchaudio, "<https://pytorch.org/audio/stable/torchaudio.html>."