INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Team02 Text-Independent Speaker Verification System for SdSV Challenge 2021

Woo Hyun Kang[1], *Nam Soo Kim*[2]

[1] Computer Research Institute of Montreal, Canada
[2]Department of Electrical and Computer Engineering and INMC,
Seoul National University, Korea

Woohyun.Kang@crim.ca, nkim@snu.ac.kr

Abstract

In this paper, we provide description of our submitted systems to the Short Duration Speaker Verification (SdSV) Challenge 2021 Task 2. The challenge provides a difficult set of cross-language text-independent speaker verification trials. Our submissions employ ResNet-based embedding networks which are trained using various strategies exploiting both in-domain and out-of-domain datasets. The results show that using the recently proposed joint factor embedding (JFE) scheme can enhance the performance by disentangling the language-dependent information from the speaker embedding. However, upon analyzing the speaker embeddings, it was found that there exists a clear discrepancy between the in-domain and out-of-domain datasets. Therefore, among our submitted systems, the best performance was achieved by pre-training the embedding system using out-of-domain dataset and fine-tuning it with only the in-domain data, which resulted in a MinDCF of 0.142716 on the SdSV2021 evaluation set.

Index Terms: speaker recognition, speech representation learning, SdSV 2021 Challenge

1. Introduction

In recent years, various methods have been proposed utilizing deep learning architectures for extracting speaker embedding vectors and have shown state-of-the-art performance when a large amount of in-domain training data is available [1, 2, 3, 4, 5, 6, 7]. However, despite their success in well-matched conditions, the deep learning-based embedding methods are vulnerable to the performance degradation caused by mismatched conditions [8].

Recently, many attempts have been made to extract an embedding vectors robust to non-speaker variability [8, 9, 10, 11]. Especially in our recent research [11], a joint factor embedding (JFE) technique is proposed where the embedding network is trained maximize the speaker-dependent information within the embedding while simultaneously maximizing the uncertainty on unwanted attributes (e.g., channel, emotion).

The SdSV Challenge provides a standard benchmark for evaluating speaker verification systems on various mismatched conditions [12]. Especially in Task 2, which consists of text-dependent speaker verification trials, the following problems should be considered:

- Only a small amount of in-domain data (i.e., DeepMine Task 2 Train Partition) is provided for training the speaker verification system.
- Task 2 consists of cross-language trials where the enrollment utterances are in Farsi and test utterances are in English.

More details about the SdSV Challenge can be found in the evaluation plan [12].

In order to solve these problems, we experimented with several training strategies including two-stage training [13] and JFE for disentangling the non-speaker information [11]. From the results, it could be seen that using the JFE scheme for language disentanglement was able to enhance the speaker verification performance. However, our analysis on the speaker embeddings showed that there exists a evident mismatch between the in-domain and out-of-domain datasets. Thus, among our experimented systems, the best performance was achieved via a two-stage optimization strategy: the network was pre-training with a large amount of out-of-domain Dataset (i.e., VoxCeleb 2) and fine-tuned with a in-domain dataset.

The contributions of this paper are as follows:

- We compare various ways to employ out-of-domain datasets (e.g., JFE, two-stage training) for the SdSV Challenge Task 2.
- We analyze the domain discrepancy between different datasets used for the SdSV Challenge Task 2.
- From the best of our knowledge, this is the first attempt on using the JFE scheme for cross-language speaker verification task.

The rest of this paper is organized as follows: The datasets used for training our systems are described in Section 2. In Section 3, detailed information on the submitted systems is depicted. Section 4 presents the results of the submitted systems and Section 5 concludes the paper.

2. Dataset

The SdSV 2021 Task 2 systems can only use a fixed training set which includes:

- VoxCeleb1 [14]: The VoxCeleb1 dataset consists of 148,642 multi-language utterances collected from 1,211 speakers.
- VoxCeleb2 [15]: The VoxCeleb2 dataset consists of 1,092,009 multi-language utterances collected from 5,994 speakers.
- LibriSpeech [16]: The LibriSpeech dataset consists of 292,367 English utterances collected from 2,484 speakers.
- Mozilla Common Voice Farsi (Mozilla) [17]: The Mozilla Common Voice Farsi dataset consists of 298,673 Farsi utterances collected from 3,654 speakers.
- DeepMine (Task 2 Train Partition) [18]: A portion of the DeepMine dataset is released as an in-domain trainingi set, which consists of Farsi utterances from 588 speakers.

http://dx.doi.org/10.21437/Interspeech.2021-249

Table 1: *Architecture for ResNetSE34, where L is the input sequence length and ASP is the attentive statistics pooling layer.*

Layer	Kernel size	Stride	Output shape
Conv1	7×7 × 16	2×1	$L/2$×64 × 16
Res1	3×3 × 16	1×1	$L/2$×64 × 16
Res2	3×3 × 32	2×2	$L/4$×32 × 32
Res3	3×3 × 64	2×2	$L/8$×16 × 64
Res4	3×3 × 128	1×1	$L/8$×16 × 128
Flatten	-	-	
ASP	-	-	4096
Linear	512	-	512

Table 2: *Architecture for ResNetSE34V2, where L is the input sequence length and ASP is the attentive statistics pooling layer.*

Layer	Kernel size	Stride	Output shape
Conv1	3×3 × 32	1×1	L×64 × 32
Res1	3×3 × 32	1×1	L×64 × 32
Res2	3×3 × 64	2×2	$L/2$×32 × 64
Res3	3×3 × 128	2×2	$L/2$×16 × 128
Res4	3×3 × 256	2×2	$L/2$×8 × 256
Flatten	-	-	
ASP	-	-	4096
Linear	512	-	512

For training the submitted systems, we used the data augmentation strategy described in [7].

The trials for Task 2 is composed of two partitions. The first partition consists of text-independent trials where the enrollment and test utterances are from the same language (Farsi). The second partition consists of cross-language trials where the enrollment utterances are in Farsi and the test utterances are in English. For the SdSV Challenge, no cross-gender trials are included.

3. System description

3.1. Backbone architecture

In our submissions, we adopted the following ResNet-34-based architectures:

- ResNetSE34 [6]: The first architecture is the Fast ResNet, which follows the same general structure with the original ResNet with 34 layers (ResNet-34) with squeeze-and-excitation, but only uses one-quarter of the channels in each residual block to reduce computational cost.

- ResNetSE34V2 [7]: The second architecture is the Half ResNet, which also follows the same structure with the ResNet-34 with squeeze-and-excitation, but uses half of the channels in each residual block.

The detailed architecture for ResNetSE34 and ResNetSE34V2 are described in Table 1 and Table 2, respectively.

In all our submitted systems, 64-dimensional mel-spectrogram acoustic features were extracted and used as input. Moreover, attentive statistical pooling (ASP) [19] layer was used to aggregate the frame-level representations, which was followed by a linear layer to obtain a 512-dimensional embedding vector.

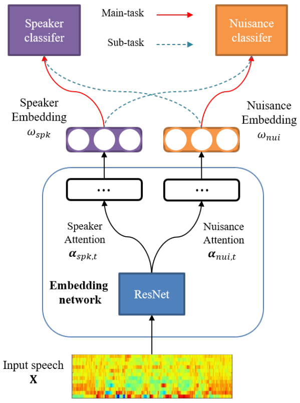
Speaker classifer
Main-task
Sub-task
Nuisance classifer
Speaker Embedding ω_{spk}
Nuisance Embedding ω_{nui}
...
...
Speaker Attention $\alpha_{spk,t}$
Nuisance Attention $\alpha_{nui,t}$
Embedding network
ResNet
Input speech X

Figure 1: *General structure of the JFE framework.*

3.2. Training objectives

3.2.1. Angular prototypical with softmax

The angular prototypical with softmax objective (Angular prototypical + Softmax) focuses on maximizing the speaker discriminability of the embedding [6]. As the name suggests, this objective involves a softmax cross-entropy loss and an angular prototypical contrastive loss to optimize the performance in terms of both speaker identification and verification. For this objective, each mini-batch consisted of 2 utterances from N speakers. When training, the embedding layer was followed by a single softmax layer where each node corresponds to different speakers in the training set. The softmax cross-entropy loss is formulated as follows:

$$L_s = -\sum_{i=1}^{2N} \log \frac{e^{W_{y_i}^T \omega_i + b_{y_i}}}{\sum_{j=1}^{C} e^{W_j^T \omega_i + b_j}}, \tag{1}$$

where C is the number of training speakers, y_i is the speaker label of embedding ω_i, and W and b are the weight and bias of the last layer, respectively.

On the other hand, the angular prototypical loss is computed as follows:

$$L_p = -\sum_{i=1}^{N} \log \frac{e^{cos(\omega_{i,1}, \omega_{j,2})}}{\sum_{j=1}^{N} e^{cos(\omega_{i,1}, \omega_{j,2})}}, \tag{2}$$

where cos is the cosine similarity operation, $\omega_{i,1}$ and $\omega_{i,2}$ are the first and second utterance of speaker i within the mini-batch, respectively. Finally, the Angular prototypical + Softmax objective is formed by adding the cross-entropy and the angular prototypical loss as,

$$L_{sp} = L_s + L_p. \tag{3}$$

3.2.2. Joint factor embedding

The JFE scheme focuses on maximizing the speaker discriminability and nuisance attribute uncertainty of the speaker embedding vector [11]. The same ResNet-based backbones described in Section 3.1 was used for the JFE framework. However, unlike the standard embedding network which only extracts a single embedding vector per utterance, the JFE framework jointly extracts a speaker embedding ω_{spk} and a nuisance embedding vector ω_{nui} as depicted in Figure 1. Therefore the

ResNet backbone was followed by two sets of ASP and linear layers to extract two 512-dimensional embedding vectors. When training, two classifiers were trained along with the embedding network:

- Speaker classifier: A single softmax layer where each node corresponds to different speakers within the training speaker where the weight and bias are W_{spk} and b_{spk}, respectively.
- Nuisance classifier: A single softmax layer where each node corresponds to different class of nuisance attribute (e.g., for language disentanglement, each node corresponds to different language) where the weight and bias are W_{nui} and b_{nui}, respectively.

In the JFE framework, each embedding is trained to perform well on their main-task, while maximizing the uncertainty on their sub-task. For example, the speaker embedding vector is trained to have maximum speaker discriminability and nuisance attribute uncertainty. On the other hand, the nuisance embedding vector is trained to have maximum nuisance discriminability and speaker uncertainty.

For this objective, each mini-batch consisted of 1 utterance from N speakers. In order to maximize the main-task discriminability, the following cross-entropy loss functions were used:

$$L_{spk} = -\sum_{i=1}^{N} \log \frac{e^{W^T_{spk,y_i}\omega_{spk,i}+b_{spk,y_i}}}{\sum_{j=1}^{C} e^{W^T_{spk,j}\omega_{spk,i}+b_{spk,j}}}, \tag{4}$$

$$L_{nui} = -\sum_{i=1}^{N} \log \frac{e^{W^T_{nui,l_i}\omega_{nui,i}+b_{nui,l_i}}}{\sum_{j=1}^{K} e^{W^T_{nui,j}\omega_{nui,i}+b_{nui,j}}}, \tag{5}$$

where K is the number of nuisance classes, and l_i is the nuisance label of $\omega_{spk,i}$ and $\omega_{nui,i}$.

On the other hand, to maximize the sub-task uncertainty, the following entropy losses were considered:

$$L_{e-spk} = -\sum_{i=1}^{N}\sum_{m=1}^{C} f_{spk,m}(\omega_{nui,i})\log f_{spk,m}(\omega_{nui,i}), \tag{6}$$

$$L_{e-nui} = -\sum_{i=1}^{N}\sum_{m=1}^{K} f_{nui,m}(\omega_{spk,i})\log f_{nui,m}(\omega_{spk,i}), \tag{7}$$

where $f_{spk,m}$ and $f_{nui,m}$ are,

$$f_{spk,m}(\omega) = \frac{e^{W^T_{spk,m}\omega+b_{spk,m}}}{\sum_{j=1}^{C} e^{W^T_{spk,j}\omega+b_{spk,j}}}, \tag{8}$$

$$f_{nui,m}(\omega) = \frac{e^{W^T_{nui,m}\omega+b_{nui,m}}}{\sum_{j=1}^{C} e^{W^T_{nui,j}\omega+b_{nui,j}}}. \tag{9}$$

Finally, the JFE objective is formed by combining the cross-entropy and entropy losses as follows:

$$L_{JFE} = L_{spk} + L_{nui} - L_{e-spk} - L_{e-nui}. \tag{10}$$

In our submissions, we considered disentangling two nuisance attributes:

- Domain disentanglement: here we attempt to disentangle the domain (dataset)-specific information from the speaker embedding by training a dataset classifier along with the speaker classifier and embedding network,

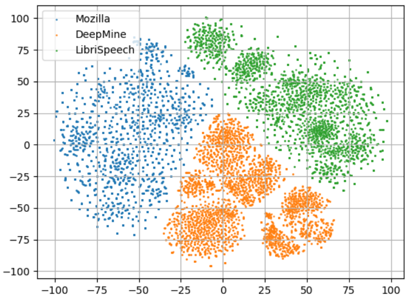

Figure 2: *T-SNE plot of the embeddings extracted using System 4. The blue, orange, green dots are the embeddings extracted from the Mozilla Common Voice Farsi, DeepMine, LibriSpeech datasets, respectively.*

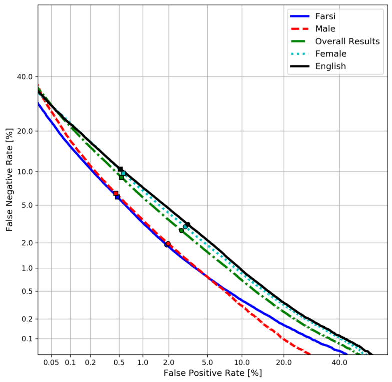

Figure 3: *DET curve for System 6.*

- Language disentanglement: here we attempt to disentangle the language-dependent information from the speaker embedding by training a language classifier along with the speaker classifier and embedding network.

For the domain disentanglement JFE, we train the system using all the available datasets (i.e., VoxCeleb1&2, LibriSpeech, Mozilla Common Voice, DeepMine), where the nuisance classifier identifies which dataset the input speech is from. On the other hand, in the language disentanglement JFE, we train the system using the Farsi datasets (i.e., Mozilla Common Voice, DeepMine) and English dataset (i.e., LibriSpeech), where the nuisance classifier identifies which language the input speech is uttering.

In all of our systems, we used ADAM optimizer to train the networks, where the initial learning rate was set to 0.0001. Exponential learning rate decay was applied, where the decay rate was 0.95. Also, each mini-batch during our training consisted of 64 utterances.

3.3. Scoring

In all our submitted systems, we used cosine similarity scoring as backend with no score normalization. Test-time augmentation (TTA) [15] was performed to extract 10 embedding vectors per utterance.

Table 3: *MinDCF results on the evaluation set*

System #	Model	Pretraining Objective	Finetuning Objective	Pretraining Dataset	Finetuning Dataset	MinDCF
System 1 (Baseline)	TDNN	Random initialization	Softmax	-	All	0.431800
System 2	ResNetSE34	Random initialization	JFE (domain disentanglement)	-	All	0.343693
System 3	ResNetSE34	Random initialization	JFE (language disentanglement)	-	DeepMine, LibriSpeech, Mozilla	0.292892
System 4	ResNetSE34V2	Random initialization	Angular prototypical + Softmax	-	VoxCeleb 2	0.309718
System 5	ResNetSE34V2	Angular prototypical + Softmax	JFE (language disentanglement)	VoxCeleb 2	DeepMine, LibriSpeech, Mozilla	0.193648
System 6	ResNetSE34V2	Angular prototypical + Softmax	Angular prototypical + Softmax	VoxCeleb 2	DeepMine	0.142716

4. Results

4.1. Comparison between JFE systems

Table 3 shows the MinDCF results of our submitted systems on the SdSV 2021 Task 2 evaluation set. System 1 is the x-vector baseline result provided by the SdSV Challenge organizers. As shown in System 2 and System 3, training the network using the JFE framework showed reasonable performance, outperforming the baseline with a relative improvement of 20.4% and 32.2%, respectively.

It is also interesting to notice that the language disentanglement JFE (System 2) performs better than the domain disentanglement JFE (System 3). This may be attributed to the extreme speaker number difference between the VoxCeleb dataset (7,205 speakers) and the in-domain dataset (i.e., DeepMine, 588 speakers).

4.2. Systems optimized with two-stage training

In order to alleviate the speaker number imbalance between different datasets, we adopted a two-stage training strategy [13], where the embedding network was first pre-trained using only VoxCeleb2 dataset (System 4), and then fine-tuned without VoxCeleb dataset (System 5, System 6). Fine-tuning the pre-trained model via language disentanglement JFE (System 5) greatly improved the performance, achieving a relative improvement of 37.5% compared to System 4. Although the JFE system showed promising performance, the best performance was achieved by fine-tuning the pre-trained model with only DeepMine dataset (System 6), which showed a relative improvement of 53.9%. This could be attributed to the domain-mismatch caused by the out-of-domain datasets used for training the JFE system (i.e., LibriSpeech, Mozilla). Such discrepancy between the in-domain and out-of-domain datasets can be noticed in Figure 2, which shows the T-SNE plot [20] of the embeddings extracted from different datasets (i.e., Mozilla, DeepMine, LibriSpeech). From Figure 2, it could be seen that there exists a clear disparity between different datasets, even if they are speaking the same language (e.g., DeepMine and Mozilla). Due to the evident between-dataset variability, systems trained on out-of-domain datasets (e.g., System 5) may not perform as well as the system trained with only the in-domain dataset (e.g., System 6).

Figure 3 depicts the DET curves of different trials for System 6. From the DET curves, it could be seen that there exists a significant performance difference between the male and female trials. Moreover, there is a huge performance gap between the Farsi and English trials. This may be attributed to the fact that the in-domain dataset (DeepMine Task2 Train Partition) contains only Farsi utterances.

5. Conclusions

In this paper, we described our submitted systems on the SdSV 2021 Challenge Task 2. The SdSV Challenge Task 2 consists of cross-language trials where the enrollment utterances are in Farsi and the test utterances are in English. Moreover, this challenge allows only a small portion of in-domain dataset for training the systems. In order to overcome these problems, we experimented with several training strategies including two-stage training and JFE for disentangling the language-dependent information from the speaker embedding. Our experimental results showed that there is a noticeable variability between different datasets used for training the systems. Among the experimented methods, the best performance was achieved by pre-training the system using a large amount of out-of-domain dataset and fine-tuning it with only the in-domain datasaet, resulting in MinDCF of 0.1427.

Although our submitted system showed great improvement over the baseline system, the results show that our system falls short for trials with English utterances. This may be attributed to the fact that the in-domain training data we employed only consists of only Farsi speech segments. In our future studies, we will focus on expanding the JFE framework to extract a language-agnostic embedding with limited target-language training set.

6. Acknowledgements

The authors wish to acknowledge funding from the Natural Sciences and Engineering Research Council of Canada (NSERC) through grant RGPIN-2019-05381. This research was supported and funded by the Korean National Police Agency. [Project Name: Real-time speaker recognition via voiceprint analysis / Project Number: PR01-02-040-17]

7. References

[1] E. Variani, E. McDermott, I. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *ICASSP*, 2014, pp. 4080–4084.

[2] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *ICASSP*, 2018, pp. 4879–4883.

[3] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: robust dnn embeddings for speaker recognition," in *ICASSP*, 2018, pp. 5329–5333.

[4] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khundanpur, "Deep neural network embeddings for text-independent speaker verification," in *Interspeech*, 2017, pp. 999–1003.

[5] W. Xie, A. Nagrani, J. S. Chung, and A. Zisserman, "Utterance-level aggregation for speaker recognition in the wild," in *ICASSP*, 2019.

[6] J. S. Chung, J. Huh, S. Mun, M. Lee, H. S. Heo, S. Choe, C. Ham, S. Jung, B.-J. Lee, and I. Han, "In defence of metric learning for speaker recognition," in *Interspeech*, 2020.

[7] H. S. Heo, B.-J. Lee, J. Huh, and J. S. Chung, "Clova baseline system for the VoxCeleb speaker recognition challenge 2020," *arXiv preprint arXiv:2009.14153*, 2020.

[8] Z. Meng, Y. Zhao, J. Li, and Y. Gong, "Adversarial speaker verification," in *ICASSP*, 2019, pp. 6216–6220.

[9] ——, "Channel adversarial training for cross-channel text-independent speaker recognition," in *ICASSP*, 2019.

[10] J. Zhou, T. Jiang, L. Li, Q. Hong, Z. Wang, and B. Xia, "Training multi-task adversarial network for extracting noise-robust speaker embedding," in *ICASSP*, 2019, pp. 6196–6200.

[11] W. H. Kang, S. H. Mun, M. H. Han, and N. S. Kim, "Disentangled speaker and nuisance attribute embedding for robust speaker verification," *IEEE Access*, vol. 8, pp. 141 838–141 849, 2020.

[12] H. Zeinali, K. A. Lee, J. Alam, and L. Burget, "Short-duration speaker verification (sdsv) challenge 2021: the challenge evaluation plan." arXiv preprint arXiv:1912.06311, Tech. Rep., 2020.

[13] H. Zeinali, S. Wang, A. Silnova, P. Matějka, and O. Plchot, "But system description to voxceleb speaker recognition challenge 2019," 2019.

[14] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," in *Interspeech*, 2017.

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

[16] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *ICASSP*, 2015, pp. 5206–5210.

[17] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *LREC*, 2020, pp. 4211–4215.

[18] H. Zeinali, J. Černocký, and L. Burget, "A multi purpose and large scale speech corpus in persian and english for speaker and speech recognition: the deepmine database," in *ASRU*. IEEE Signal Processing Society, 2019, pp. 397–402. [Online]. Available: https://www.fit.vut.cz/research/publication/12153

[19] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive statistics pooling for deep speaker embedding," in *Interspeech*, 2018, pp. 2252–2256.

[20] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008. [Online]. Available: http://jmlr.org/papers/v9/vandermaaten08a.html