Veridas Solution for SdSV Challenge Technical Report

Guillermo Barbadillo, Santiago Prieto

Veridas

{gbarbadillo, sprieto}@das-nano.com

Abstract

In this report, we describe the submission of Veridas Digital Authentication Solutions S.L. team for the Short-duration Speaker Verification (SdSV) Challenge 2020. Veridas team achieved the 3rd position in the challenge among more than 60 teams from all over the world. Submitted system is a fusion of 6 Convolutional Neural Network (CNN) models. All networks have a ResNet34 architecture and use two-dimensional CNNs.

Index Terms: speaker recognition, short duration, Convolutional Neural Network (CNN).

1. Introduction

In this report, we describe the solution of Veridas Digital Authentication Solutions S.L. team for the Short-duration Speaker Verification (SdSV) Challenge 2020. This challenge is based on DeepMine Database which is a speech database in Persian and English designed to build and evaluate text-dependent, text-prompted, and text-independent speaker verification, as well as Persian speech recognition systems[1].

The evaluation plan[2] describes the conditions of the challenge such as:

- fixed training data
- normalized minimum Detection Cost Function as metric
- cross-language trials

This work is built and greatly inspired by the solution of Brno University of Technology (BUT) team to the VoxCeleb Speaker Recognition Challenge (VoxSRC) 2019 [3]. Although we experimented with different ResNet and TDNN architectures the best results were obtained using variants of the architecture proposed on BUT solution.

The rest of the document is organized as follows: in Section 2, we first describe the setup for the challenge. In Section 3, the systems based on ResNet34 DNN will be explained. Backends and fusion strategies are outlined in Section 4 and finally the results and analysis are presented in Section 5.

2. Experimental Setup

2.1. Training data, Augmentations

For training the following datasets were used:

- Voxceleb-1 dev
- Voxceleb-2 dev
- Librispeech train
- DeepMine train (folds 1-9)

The DeepMine train dataset was randomly splitted in 10 folds. One was used for validation and the other nine were used for training.

Data augmentation was based on Kaldi recipe[4]. The categories for data augmentation were:

- Reverberation with RIRs[5]
- Augmentation with Musan[6] noise
- Augmentation with Musan music
- Augmentation with Musan bable

For training splits of 3 seconds of audio were used. Speakers with less than 200 seconds of audio were discarded.

2.2. Validation datasets

For validation the following datasets were used:

- DeepMine train (fold 0)
- Voxceleb-1 test
- Voxceleb-2 test

  The DeepMine train dataset was randomly splitted in 10 folds. One was used for validation and the other nine were used for training.

2.3. Input features

All the models in the final system used Log Mel features as input. The number of filters was modified to force variability between models. 60, 80 and 100 log mel filters were used.

The audios were cleaned using an energy-based VAD.

3. DNN based Systems

3.1 ResNet34

All the models used on the final solution were based on ResNet34 architecture as described in the BUT solution to VoxCeleb challenge[3].
The differences between the models are highlighted in table 1.

Table 1: *Differences between the models.*

Model	mel filters	stats	parameters
1	80	mean	9M
2	80	mean + stddev	9M
3	100	mean	9M
4	100	mean + stddev	9M
5	80	mean + stddev	13M
6	60	mean + stddev	9M

3.2 Additive angular margin loss

As proposed in the paper ArcFace: Additive Angular Margin Loss for Deep Face Recognition[7] m2 margin was used in all the models.

3.3 Fine-tuning with DeepMine dataset

After training with all the datasets the models were fine tuned using only DeepMine dataset. This step was crucial because DeepMine dataset is small compared to the other datasets used and thus has a lower weight when being trained along them. This fine-tuning could improve up to 1% in EER and 0.05 in minDCF of our internal validation dataset.

4. Backend

4.1 Euclidean distance

The embeddings of the speakers are restricted to lie on a hypersphere of radius 1. Following that restriction using euclidean distance is equivalent to cosine distance but euclidean distance is faster to compute. Thus we used euclidean distance for measuring similarity between embeddings.

There was no preprocessing nor centering of the embeddings.

4.2 Score normalization

We used adaptative symmetric score normalization (adapt S-norm)[8] with 250 top scoring speakers. The cohort was created using all training speakers.

An improvement in minDCF of around 0.002 was observed when applying this score normalization.

4.3 Calibration and fusion

The final system is a simple average of the predictions of the best 6 individual models.

More advanced ensembling techniques were tried but due to the weak correlation between our internal validation set and the public leaderboard no gains were obtained.

5. Results and Analysis

Table 2: *Results of the different models.*

Model	Public LB EER	Public LB MinDCF	Private LB EER	Private LB MinDCF
1	1.94	0.0856	1.928	0.0852
2	2.002	0.0865	1.975	0.0858
3	1.923	0.0839	1.916	0.0840
4	2.078	0.0886	2.063	0.0883
5	1.905	0.0841	1.887	0.0840
6	2.151	0.0946	2.132	0.0939
ensemble	1.765	0.0769	1.744	0.0765

The final submission was an ensemble of 6 models. The predictions were simply averaged to make the ensemble. It is worth noting that at the time of making the prediction the public scores of the individual models were not known because of the limitation of 1 submission per day. Maybe removing the model 6 from the ensemble may improve the score slightly.

This ensemble gave Veridas the third position in the challenge as it is shown on table 3.

Table 3. Final positions of the challenge

position	team	MinDCF
1	IDLab	0.0651
2	NICT	0.0740
3	**Veridas**	**0.0765**
4	Team 64	0.0836
5	Team05	0.0951
6	JHU	0.1051
7	TJU-cca	0.1118
8	TalTech	0.1178
9	ID R&D	0.1246
10	CSTR	0.1256

It is very remarkable that if the best single model had been chosen for the final submission Veridas would have achieved the 4 position very close to the 3.

6. References

[1] H. Zeinali, H. Sameti, and T. Stafylakis, “DeepMine speech processing database: Text-dependent and independent speaker verification and speech recognition in Persian and English.” in *Proc. Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018, pp. 386–392.

[2] Zeinali, Hossein nad Lee, Kong Aik and Alam, Jahangir and Burget, Luka “Short-duration Speaker Verification (SdSV) Challenge 2020: the Challenge Evaluation Plan”

[3] Hossein Zeinali, Shuai Wang, Anna Silnova, Pavel Matějka, Oldřich Plchot “BUT System Description to VoxCeleb Speaker Recognition Challenge 2019”

[4] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, Y. Qian, P. Schwarz, J. Silovský, G. Stemmer, and K. Vesel, “The kaldi speech recog-nition toolkit,” 01 2011.

[5] RIR_NOISES, http://www.openslr.org/28

[6] MUSAN, http://www.openslr.org/17

[7] Jiankang Deng, Jia Guo, Niannan Xue, Stefanos Zafeiriou “ArcFace: Additive Angular Margin Loss for Deep Face Recognition”

[8] Pavel Matejka, Ond ˇ rej Novotn ˇ y, Old ´ rich Plchot, Luk ˇ a´s Burget, ˇMireia Diez Sanchez, Jan “Honza” ´ Cernock ˇ y´ “Analysis of Score Normalization in Multilingual Speaker Recognition”