OPEN-SET SPEAKER RECOGNITION WITH AUGMENTED I-VECTORS

Kevin Wilkinghoff

Fraunhofer Institute for Communication, Information Processing and Ergonomics FKIE
Fraunhoferstraße 20, 53343 Wachtberg, Germany
kevin.wilkinghoff@fkie.fraunhofer.de

ABSTRACT

This paper contains a description of a speaker recognition system submitted to the "1st Multi-target speaker detection and identification Challenge Evaluation". For both stages, detection and identification, the system consists of two differently trained models. First, Linear Discriminant Analysis (LDA) in combination with a Support Vector Machine is applied to determine whether a given file belongs to one of the blacklist speakers. Secondly, a Neural Network which improves the discriminative behavior of the i-vectors as well as LDA are used to decide which blacklist speaker is present in the utterance, in case one of the blacklist speakers has been detected before. Our system significantly improves upon the baseline system since both equal error rates (EERs) obtained with the development set are reduced. The top-S EER is improved from 2.01% to 1.1% and the top-1 EER from 12.26% to 8.18%. In additional experiments it is shown that using the Neural Network for preprocessing is beneficial regardless of the model being used for the actual identification.

Index Terms— speaker recognition, open-set recognition, i-vector, deep learning, cosine similarity

1. INTRODUCTION

The "1st Multi-target speaker detection and identification Challenge Evaluation" (MCE 2018) [1] is targeted at doing open-set speaker recognition with i-vectors [2]. This means that a speaker recognition system first needs to decide whether an utterance, represented by an i-vector, belongs to one of the target speakers called blacklist speakers. In a second step, the system needs to determine to which of the blacklist speakers the i-vector belongs to. The difficulty of this task is that there are not only known speakers and known unknown speakers, which are all available when training, but also unknown unknown speakers (see [3]). Thus, the speaker recognition system needs to be able to also discriminate the known speakers against all possible unknown speakers whose data is not available during training. This is a more realistic scenario for all practical speaker recognition applications and makes MCE 2018 a very interesting challenge.

The concept of the baseline system provided by the organizers of the challenge is described in [4] and uses simple cosine similarity in combination with Multi-Target Score Normalization. Other works related to open-set speaker recognition mainly focus on score normalization techniques [5, 6] as well. A problem one faces when using score normalization techniques is that one can only obtain meaningful scores with data that has not been used for training discriminative models. The reason is that, inherently, the performance is usually close to perfect on the training data and the corresponding scores are also almost ideal. Furthermore, it should be avoided to normalize the scores using multiple test files as this would destroy the independence between individual trials. In conclusion, one needs to keep data solely for the purpose of obtaining meaningful scores. But as the number of samples per speaker is often very limited in practice, which is also the case for this challenge, using all data for training better performing models is preferable and leads to better results. This is the reason why we did not use any score normalization technique in our proposed speaker recognition system.

This paper is organized as follows: First, we describe our proposed speaker recognition system in detail. It basically consists of two different models for the detection and identification stage which are both based on Linear Discriminant Analysis (LDA). For detecting the blacklist speakers we use a Support Vector Machine (SVM) [7] with Radial Basis Function (RBF) kernels. When identifying which blacklist speaker is present, a Neural Network is used for preprocessing the i-vectors before applying LDA. Afterwards, we use simply cosine similarity to obtain the scores and pick the speaker corresponding to the largest score. In the next section, the performance of our system is compared to the one of the baseline system. Additionally, it is shown that preprocessing the i-vectors does indeed help to lower the equal error rates.

2. SYSTEM DESCRIPTION

The general structure of our speaker recognition system can be found in Fig. 1. Its fundamental idea is to use two different models in the detection and identification stage. For detecting the blacklist speakers, first an LDA model is trained which discriminates between the two speaker classes "black-

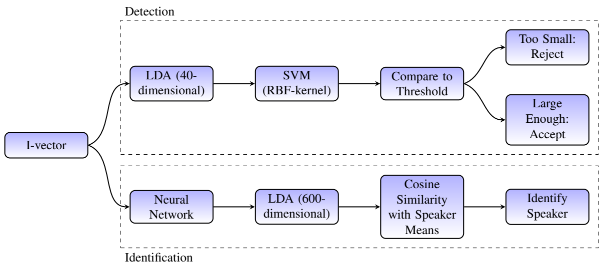

Fig. 1. Structure of the speaker recognition system

list" and "non-blacklist". Additionally, the dimension of the i-vectors is reduced to 40. This particular dimension has been chosen because it yielded the lowest EER when evaluating on the development set. Then, a two-class SVM with RBF-kernels is trained using the parameter settings $C = 1$ and $\gamma = 0.01$. Again, both values have been tuned by evaluating the performance on the development set. This SVM also outputs probabilities for each i-vector stating how likely it is that this i-vector belongs to any of the blacklist speakers. As it is done with the baseline model, these probabilities are then utilized as scores to either accept or reject an i-vector by using a threshold.

For preprocessing the i-vectors in the identification stage, we used a Neural Network whose structure is shown in Tab. 1. Its purpose is to increase the number of training samples by using a simple data augmentation technique we designed for i-vectors. This is motivated by the fact that there are only very few samples per class, 3 in the training set and 1 in the development set. Hence, the Neural Network aims to learn a projection of the i-vectors which is robust to missing information and can cope with additional variance introduced to the i-vectors. We used a combination of Batch Normalization [8], Dropout [9] and L2-normalization to increase the number of training samples. As all i-vectors will be evaluated using the Cosine Similarity, they will also be projected to the unit sphere. Thus, normalizing them with respect to the Euclidean norm (L2-normalization) does not harm the information contained in the i-vectors. Also note that the combination of Dropout and L2-normalization is a very effective and computationally cheap way of augmenting the i-vectors. The reason is that dropping a different number of randomly chosen dimensions also changes the values of the remaining entries differently in each epoch due to the normalization. The Neural Network has been implemented with Keras [10] and Tensorflow [11]. It has been trained for 2500 epochs with a batch size of 128 by minimizing the Cosine Proximity loss via Adam [12] with a learning rate of 0.001 and a weight decay of 0.0001. The size of the first layer is higher (1500) than the i-vector dimension (600) because there may be samples that are only linear separable in the higher dimensional vector space. We also experimented with applying nonlinear functions in the dense layers instead but this always led to much worse performance. Furthermore, using more layers did not result in lower equal error rates.

Table 1. Architecture of the Neural Network used for preprocessing the i-vectors.

Layer	Output Shape	#Parameters
Input	600	0
Dense (Linear)	1500	901,500
Batch Normalization	1500	6,000
Dropout (0.5)	1500	0
L2-normalization	1500	0
Dense (Linear)	600	900,600
Batch Normalization	600	2,400
Dropout (0.5)	256	0
L2-normalization	600	0
Cosine Similarity	3631	0

Table 2. Comparison of top-1 equal error rates obtained with and without augmenting the i-vectors on the development set.

model	without augmentation	with augmentation
cosine similarity	12.28% (444 errors)	10.42% (375 errors)
LDA	8.34% (299 errors)	**8.18% (293 errors)**
PLDA	10.30% (372 errors)	10.06% (363 errors)

Next, LDA is applied to the augmented i-vectors but in contrast to the detection stage the dimension is not reduced. For each blacklist speaker, the mean of all augmented and LDA-projected i-vectors belonging to that speaker is taken. When testing, the test i-vector is processed in the same way and then compared to all speaker-specific i-vectors via cosine-similarity. Finally, the blacklist speaker is identified by returning the one resulting in the highest similarity.

3. EXPERIMENTAL EVALUATIONS

First, we trained the LDA and SVM models for the detection phase with [13] as described in Section 2. Using the results of the detection phase obtained with our system, we then evaluated multiple models on the development set. More concretely, we compared using cosine similarity for identification, LDA or Probabilistic Linear Discriminant Analysis (PLDA) as implemented in [14] (trained for 20 iterations with latent variable dimensions of size 500 each). Additionally, each model has been evaluated with and without augmenting all i-vectors with the Neural Network before. All results can be found in Table 2. It can be seen that LDA and PLDA both help to decrease the EERs but LDA performs much better than PLDA. Therefore, LDA is used in our final system. Furthermore, augmenting the i-vectors with the Neural Network always improves the performance although the effect is noticeably higher when using cosine similarity. In conclusion, it is a very helpful preprocessing step which we included in our system.

In Fig. 2, we compared the EERs obtained with our sys-

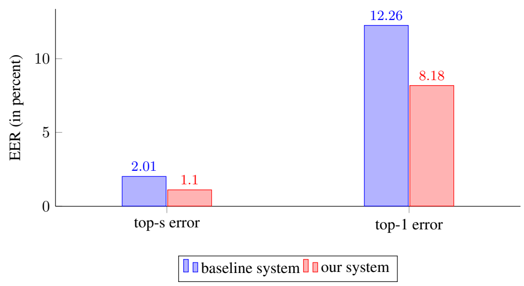

Fig. 2. Equal error rate of our system compared to the baseline system.

tem to the ones of the baseline system. It is immediately visible that our system performs much better as the top-S EER is almost halved from 2% to 1.1% and the top-1 EER is decreased by one third from 12.26% to 8.18%. Note that the results have been computed by training all models with the training set and evaluating them with the development set. For our final submitted system, we utilized the development set as additional training data which hopefully improved the performance even more. We also experimented with ensembling multiple models but decided to only use single models since ensembles did not lead to significant improvements over the best performing single models. Furthermore, it kept the model as simple as it is which makes it very easy to replicate the results.

4. CONCLUSION AND FUTURE WORK

In this paper, we presented a system for doing open-set speaker recognition with i-vectors which has been submitted to MCE 2018. The system is based on two different models both based on LDA. For the detection stage, we used LDA in combination with RBF-kernel SVMs. In case a blacklist speaker has been detected, a Neural Network for preprocessing as well as LDA are applied and the speaker is identified via cosine similarity. In experiments conducted with the development set, it has been shown that the system performs much better than the baseline system. Additionally, using the Neural Network for preprocessing the i-vectors in the identification stage lead to improved equal error rates regardless of the model.

In the near future, we plan to examine the following augmentations of the presented system possibly leading to an even better performance: One of them is to apply Deep LDA [15] instead of LDA in the identification stage. It can be attached as an additional layer to the Neural Network preprocessing the i-vectors. Thus, both steps can be trained jointly in a single Neural Network which possibly leads to a further improvement of the performance. Furthermore, we want to experiment with other losses as the triplet-based loss [16] or the one presented in [17]. An enhancement in the detection phase may be accomplished by using discriminative PLDA with SVMs as presented in [18, 19] or Weibull-calibrated SVMs [3] which both seem to be worth investigating.

5. REFERENCES

[1] Suwon Shon, Najim Dehak, Douglas Reynolds, and James Glass, "Mce 2018: The 1st multi-target speaker detection and identification challenge evaluation (mce) plan, dataset and baseline system," *ArXiv e-prints arXiv:1807.06663*, 2018.

[2] Najim Dehak, Patrick J. Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[3] Walter J. Scheirer, Lalit P. Jain, and Terrance E. Boult, "Probability models for open set recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 11, pp. 2317–2324, 2014.

[4] Elliot Singer and Douglas A. Reynolds, "Analysis of multitarget detection for speaker and language recognition," in *ODYSSEY - The Speaker and Language Recognition Workshop*, 2004, number 4, pp. 301–308.

[5] J. Fortuna, Perasiriyan Sivakumaran, Aladdin M. Ariyaeeinia, and Amit Malegaonkar, "Relative effectiveness of score normalisation methods in open-set speaker identification," in *ODYSSEY - The Speaker and Language Recognition Workshop*, 2004.

[6] Yaniv Zigel and Moshe Wasserblat, "How to deal with multiple-targets in speaker identification systems?," in *ODYSSEY - The Speaker and Language Recognition Workshop*. IEEE, 2006, pp. 1–7.

[7] Corinna Cortes and Vladimir Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[8] Sergey Ioffe and Christian Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning*, 2015, vol. 37, pp. 448–456.

[9] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[10] François Chollet et al., "Keras," https://keras.io, 2015.

[11] Martin Abadi et al., "Tensorflow: a system for large-scale machine learning," *OSDI*, vol. 16, pp. 265–283, 2016.

[12] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[14] Aleksandr Sizov, Kong Aik Lee, and Tomi Kinnunen, "Unifying probabilistic linear discriminant analysis variants in biometric authentication," in *Proc. S+SSPR*. Springer, 2014, pp. 464–475, Software available at https://sites.google.com/site/fastplda/.

[15] Matthias Dorfer, Rainer Kelz, and Gerhard Widmer, "Deep linear discriminant analysis," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

[16] Alexander Hermans, Lucas Beyer, and Bastian Leibe, "In defense of the triplet loss for person re-identification," *arXiv preprint arXiv:1703.07737*, 2017.

[17] Nicolai Wojke and Alex Bewley, "Deep cosine metric learning for person re-identification," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 748–756.

[18] Sandro Cumani, Niko Brümmer, Lukáš Burget, and Pietro Laface, "Fast discriminative speaker verification in the i-vector space," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 4852–4855.

[19] Lukáš Burget, Oldřich Plchot, Sandro Cumani, Ondřej Glembek, Pavel Matějka, and Niko Brümmer, "Discriminatively trained probabilistic linear discriminant analysis for speaker verification," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 4832–4835.