Open access • Proceedings Article • DOI:10.21437/INTERSPEECH.2020-2662

Cross-lingual speaker verification with domain-balanced hard prototype mining and language-dependent score normalization — Source link

Jenthe Thienpondt, Brecht Desplanques, Kris Demuynck

Institutions: Ghent University

Published on: 25 Oct 2020 - Conference of the International Speech Communication Association

INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Cross-Lingual Speaker Verification with Domain-Balanced Hard Prototype Mining and Language-Dependent Score Normalization

Jenthe Thienpondt, Brecht Desplanques, Kris Demuynck

IDLab, Department of Electronics and Information Systems, Ghent University - imec, Belgium
jenthe.thienpondt@ugent.be, brecht.desplanques@ugent.be

Abstract

In this paper we describe the top-scoring IDLab submission for the text-independent task of the Short-duration Speaker Verification (SdSV) Challenge 2020. The main difficulty of the challenge exists in the large degree of varying phonetic overlap between the potentially cross-lingual trials, along with the limited availability of in-domain DeepMine Farsi training data. We introduce domain-balanced hard prototype mining to fine-tune the state-of-the-art ECAPA-TDNN x-vector based speaker embedding extractor. The sample mining technique efficiently exploits speaker distances between the speaker prototypes of the popular AAM-softmax loss function to construct challenging training batches that are balanced on the domain-level. To enhance the scoring of cross-lingual trials, we propose a language-dependent s-norm score normalization. The imposter cohort only contains data from the Farsi target-domain which simulates the enrollment data always being Farsi. In case a Gaussian-Backend language model detects the test speaker embedding to contain English, a cross-language compensation offset determined on the AAM-softmax speaker prototypes is subtracted from the maximum expected imposter mean score. A fusion of five systems with minor topological tweaks resulted in a final MinDCF and EER of 0.065 and 1.45% respectively on the SdSVC evaluation set.

Index Terms: speaker recognition, cross-lingual speaker verification, x-vectors, SdSV Challenge 2020

1. Introduction

Speaker verification systems have improved significantly by the strength of deep learning [1, 2] and the increase in publicly available labeled training data [3, 4]. However, most of these datasets tend to focus on the Anglosphere, making it hard to produce speaker embeddings that perform well on out-of-domain data.

The SdSV Challenge uses this notion to create a challenging set of speaker verification trials, divided in two separate tasks. Task 1 consists of text-dependent speaker verification, for which both the lexical content and speaker identity should be equal across the enrollment and test utterances to indicate a valid trial. Task 2 is concerned with text-independent speaker verification, which only takes the speaker identities into account. This paper focuses solely on our submission to the second text-independent task.

Task 2 systems can only use a fixed training dataset consisting of VoxCeleb1 [3], VoxCeleb2 [4], LibriSpeech [5] and a part of the DeepMine corpus [6] containing in-domain Farsi training utterances across 588 speakers. Trials consists of producing a speaker similarity score between multiple Farsi enrollment utterances and a test utterance. The test utterance can either contain Farsi or English speech. Consequently, speaker verification systems should be able to reduce the language bias in cross-lingual trials. More details about the SdSV Challenge conditions can be found in the evaluation plan [7].

The rest of the paper is organized as follows: Section 2 will describe the IDLab SdSVC final submission. The state-of-the-art ECAPA-TDNN [8] architecture is combined with adapted training procedures and backend scoring to tackle the challenge-specific difficulties. It is followed by a more in-depth analysis of the proposed approach in Section 3. Section 4 will give the concluding remarks.

2. SdSVC IDLab submission

This section is a system description of the IDLab SdSVC final submission. We start with a single system ECAPA-TDNN baseline [8]. The subsequent sections will tackle the problems of domain adaptation and cross-lingual language effects present in the SdSV Challenge data. The final subsection discusses system fusion.

2.1. The ECAPA-TDNN baseline system

All submitted speaker verification systems make use of the ECAPA-TDNN architecture proposed in [8]. This architecture is based on the well-known x-vector topology [1] and introduces several enhancements to extract more robust speaker embeddings. It incorporates Squeeze-Excitation (SE) blocks [9], multi-scale Res2Net [10] features, multi-layer feature aggregation [11] and channel-dependent attentive statistics poolings [8]. The network topology is shown in Figure 1. Implementation details and performance analysis of this architecture can be found in [8]. We deviate slightly from the original architecture by also incorporating SE-Blocks in the residual connections.

We use all allowed training data, except the VoxCeleb1 test partition and LibriSpeech, for which only the *train-other-500* subset [5] is considered. This amounts to 9077 training speakers. We create 9 additional augmented copies of the training data following the Kaldi recipe [2] in combination with the MUSAN corpus (babble, noise, music) [12] and the RIR[13] dataset (reverb). The remaining augmentations are generated with the open-source SoX (tempo up, tempo down, phaser and flanger) and FFmpeg (alternating opus and aac compression) libraries.

To avoid overfitting during the ECAPA-TDNN training process, we take a random crop of 2 to 3 seconds of the utterances during each iteration. Similarly, we incorporate SpecAugment [14] as an online augmentation method which randomly masks 0 to 5 time frames and 0 to 8 frequency bands of the training log mel-spectrograms. The input features are 64-dimensional MFCCs from a 25 ms window with a 10 ms frame shift. The MFCCs are normalized through cepstral mean subtraction and no voice activity detection is applied.

We use the Angular Additive Margin (AAM) softmax [15] as training criterion for the model. The system is trained with the Adam optimizer [16] until convergence on a small SdSVC

http://dx.doi.org/10.21437/Interspeech.2020-2662

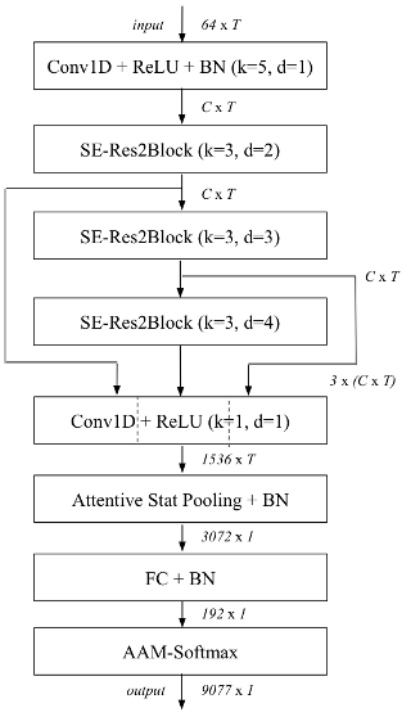

Figure 1: *Network topology of the ECAPA-TDNN. We denote k for kernel size and d for dilation spacing of the Conv1D layers or SE-Res2Blocks. C and T correspond to the channel and temporal dimension of the intermediate feature-maps respectively.*

validation subset that contains about 2.5% of the Farsi training utterances. The training protocol uses a cyclical learning rate schedule with the *triangular2* policy [17]. The learning rate is varied between a minimum of 1e-8 and decaying maximum of 1e-3 during cycles of 130k iterations. A weight decay of 2e-5 is applied on all weights of the model except for the AAM-softmax layer which uses a weight decay value of 2e-4. We use a mini-batch size of 128.

The speaker enrollment models are constructed by averaging the corresponding L_2-normalized enrollment embeddings produced by the final fully-connected layer of the ECAPA-TDNN. The verification trials are scored by calculating the cosine distance between the enrollment model and the test utterance embedding. Scores are normalized using top-40 adaptive s-normalization [18, 19]. The imposter cohort consists of speakers represented by the average of all their length-normalized training embeddings. The final scores are calibrated with logistic regression [20] on our small SdSVC validation subset.

We consider five implementations with minor topological differences as shown in Table 1. We alternate the embedding size between 192 and 256. The Res2Net multi scale features inside the SE-Res2Blocks are optionally replaced by the standard TDNN 1-dimensional dilated convolutions. *Summed* indicates if the input of each SE-Res(2)Block is the sum of the output of all preceding SE-Res(2)Blocks instead of only considering the output of the preceding block. The number of filters in the convolutional frame layers C is set to 1024, which is reduced to 512 in the bottleneck of the SE-Res(2)Blocks to limit the amount of model parameters. However, system 5 is developed without this constraint and the channel dimension is kept to 2048 for all feature maps in the frame layers.

2.2. Hard prototype mining

To further improve performance on the baseline, we investigate how to exploit the information of the in-domain training data more efficiently. We combine targeting harder samples and putting more importance to target-domain samples with our proposed Hard Prototype Mining (HPM) fine-tuning strategy.

Hard negative mining in speaker recognition systems has mostly been explored in conjunction with metric learning objective functions [21, 22, 23]. A current overview of these loss functions applied within speaker recognition is provided in [24]. Metric learning objectives shift a lot of implementation challenges to the sample mining process. In contrast, HPM is a simple and computationally efficient hard negative mining method that interoperates with the AAM-softmax loss.

2.2.1. Broad hard prototype mining

The general principle behind HPM is to detect hard speakers that confuse the speaker verification system the most and to subsequently construct batches with utterances from these speakers. A direct and continuous measurement of speaker confusion between all training samples would be computationally infeasible. Hence, we need an approximate and efficient way to compute training speaker similarities that can be easily updated as the training progresses.

We interpret the weights of the AAM-softmax layer as approximations of the class-centers of the training speakers and refer to them as speaker prototypes. As these trainable weights are already a part of the model, there are no additional computations needed. Given batch size n and N training speakers, the AAM-softmax loss L with margin m is defined as:

$$L = -\frac{1}{n}\sum_{i=1}^{n} log\frac{e^{s(cos(\theta_{y_i}+m))}}{e^{s(cos(\theta_{y_i}+m))} + \sum_{j=1,j\neq y_i}^{N} e^{s(cos(\theta_j))}} \quad (1)$$

where θ_{y_i} is the angle between the sample embedding $\boldsymbol{x}_i$ with corresponding speaker identity y_i and the speaker prototype $\boldsymbol{W}_{y_i}$. θ_j is the angle with all other L_2-normalized speaker prototypes stored in a trainable matrix $\boldsymbol{W} \in \mathbb{R}^{D\times N}$ with D indicating the embedding size. A speaker similarity matrix $\boldsymbol{S} \in \mathbb{R}^{N\times N}$ can be constructed from $\boldsymbol{W}^T\boldsymbol{W}$, containing the cosine distances between all pairs of speaker prototypes.

A straightforward way of constructing batches would be to only mine samples from the most difficult speaker pairs according to $\boldsymbol{S}$. However, this could lead to oversampling a narrow group of speakers which potentially degrades generalization performance. Consequently, we construct mini-batches by iterating randomly over all N training speakers. Each iteration determines S speakers, irrespective of their similarity, for which U random utterances are sampled from each of their I most similar speakers, including the selected speaker. This implies that $S \times U \times I$ should be equal to the batch size n. When we have iterated over all training speakers, the similarity matrix $\boldsymbol{S}$ is updated and the batch generating process is repeated. Experiments indicate that given a batch size of 128, $S = 16$, $I = 8$ and $U = 1$ result in good performance.

To fine-tune all models in this paper, we reduce the maximum of the cyclical learning rate to 1e-4 and reduce the cycle length to 60k iterations.

2.2.2. Domain-balanced hard prototype mining

In the general HPM strategy discussed above, the S selected speakers are randomly sampled from all N training speakers.

Table 1: *EER and MinDCF performance of all individual systems and final fusion on the VoxCeleb1 and SdSVC 2020 test sets. All HPM models use our domain-balanced hard prototype mining technique as explained in Section 2.2.2. LID denotes usage of our language-dependent s-normalization variant introduced in Section 2.3.*

#	System (# params)	Emb. dim	Res2	Summed	Fine-tune	VoxCeleb1		SdSVC 2020	
						EER(%)	**MinDCF**	**EER(%)**	**MinDCF**
1	ECAPA-TDNN (24M)	192	no	no	baseline	0.94	0.1181	2.38	0.1042
					HPM	0.85	0.0945	1.81	0.0798
					HPM + LID	-	-	1.75	0.0781
2	ECAPA-TDNN (24M)	192	no	yes	baseline	1.03	0.1260	2.34	0.0996
					HPM	0.96	0.1248	1.77	0.0791
					HPM + LID	-	-	1.72	0.0775
3	ECAPA-TDNN (16M)	256	yes	no	baseline	0.86	0.0969	2.32	0.1008
					HPM	0.81	0.1033	1.75	0.0784
					HPM + LID	-	-	1.69	0.0764
4	ECAPA-TDNN (16M)	256	yes	yes	baseline	0.88	0.1101	2.32	0.0994
					HPM	0.88	0.1161	1.69	0.0759
					HPM + LID	-	-	1.63	0.0742
5	ECAPA-TDNN (44M)	256	yes	yes	baseline	0.87	0.0824	2.13	0.0938
					HPM	0.79	0.1010	1.69	0.0759
					HPM + LID	-	-	**1.63**	**0.0739**
	Weighted fusion of 1-5				HPM + LID	-	-	**1.45**	**0.0651**

However, there are only 588 in-domain Farsi speakers out of a total of 9077 training speakers. This bias possibly leads to speaker embeddings that are sub-optimal towards the target-domain. A common transfer learning technique is to fine-tune a pre-trained model on the target-domain data with the goal to correct the data distribution mismatch between the training and target-domain. Due to the tendency of neural networks to easily overfit on small datasets, we opt to learn a robust embedding that performs reasonably well on both the available out-of-domain VoxCeleb data and target-domain DeepMine training data.

We correct the bias towards the VoxCeleb and LibriSpeech corpus by equalizing the sample probability for each domain. During the construction of the batches, subsequent selections of the S speakers cover a set of all 588 Farsi speakers and 588 random speakers from both the VoxCeleb and LibriSpeech domain. When the set runs empty, the similarity matrix $\boldsymbol{S}$ is updated and 588 new speakers are randomly selected from the out-of-domain data to allow reiteration of the batch generation process. This process assigns more importance towards samples from hard speakers in the target-domain, while still allowing the network to learn from samples of challenging out-of-domain speakers.

2.3. Adaptive s-normalization with language offset

Based on [25], we set the imposter cohort of the adaptive s-normalization to contain in-domain Farsi data only. However, an unknown portion of the test utterances in the SdSVC trials is English. In case of a speaker verification trial with language mismatch, this will result in an overestimated mean imposter score for the Farsi enrollment model, as it will only be compared against Farsi imposters. We introduce a language-dependent offset in the adaptive s-normalization procedure to compensate for this effect.

Given a trial score $s(e,t)$ between the enrollment model e and test utterance t, the language-dependent s-normalized score is defined as:

$$s(e,t)_n = \frac{s(e,t) - \mu(S_t)}{\sigma(S_t)} + \frac{s(e,t) - (\mu(S_e) - \alpha)}{\sigma(S_e)}. \quad (2)$$

with S_i the set of scores of the speaker embedding i against its top-N imposter cohort, with $\mu(S_i)$ the mean of those scores and $\sigma(S_i)$ the standard deviation. α is the language-dependent compensation offset. It is defined as zero if there is no language mismatch detected and in that case regular adaptive s-norm is applied. When during test time the test utterance is detected to be English, we enable the language offset. Given $\mu_{S_{FA}}$ as the expected mean imposter score of Farsi imposters against a Farsi speaker and $\mu_{S_{USA}}$ as the expected mean imposter score of USA-English imposters against a Farsi speaker, we define this compensation offset α as $\mu_{S_{FA}} - \mu_{S_{USA}}$. The mean imposter values can be easily estimated on the speaker prototypes stored in the AAM-softmax module by applying s-norm on the relevant prototypes.

To detect the language of the test utterance given its embedding, we train a Language Identification (LID) module based on a Gaussian Backend (GB) [26] modeled on the L_2-normalized AAM speaker prototypes of the Farsi and the USA speakers. However, there will be a mismatch between the English spoken by a native Farsi speaker and a USA citizen. To compensate for this effect we interpolate between the GB mean vector for the USA language class $\boldsymbol{\mu}_{USA}$ and the mean vector corresponding with Farsi $\boldsymbol{\mu}_{FA}$ and set the expected mean embedding of the English model to $0.75\boldsymbol{\mu}_{USA} + 0.25\boldsymbol{\mu}_{FA}$. This adapted language model should be able to robustly detect English spoken by a native Farsi speaker.

2.4. Final submission

The IDLab final submission for the SdSVC consists of a fusion of the five proposed ECAPA-TDNN systems fine-tuned with

domain-balanced HPM combined with language-dependent s-normalization with the LID labels extracted from System 1. The fusion is realized on the score level by taking a weighted average over the calibrated scores of each individual system. The systems that incorporate Res2 modules are given double the weight in the averaging compared to the other systems.

3. Results and additional experiments

3.1. ECAPA-TDNN baseline performance

The baseline performance of the ECAPA-TDNN architectures on the SdSVC evaluation data is shown in Table 1. We also keep track of results on the original VoxCeleb1 test set to verify the system is not overfitting on the training data. No s-normalization is used for the VoxCeleb1 evaluation results.

These baseline single system implementations show strong and similar performance on both the SdSVC and VoxCeleb data, reaching up to an EER and MinDCF of 2.13% and 0.0938 respectively on the SdSVC test set. System 4 with SE-Res2Blocks and summed inputs slightly outperforms the other equally sized systems, while its much larger counterpart System 5 only delivers a small performance gain.

3.2. Domain-balanced HPM fine-tuning

The impact of domain-balanced HPM fine-tuning on the baseline systems can be found in Table 1. After fine-tuning, all systems perform significantly better on the SdSVC test set with an average improvement of 24.1% in EER and 21.8% in MinDCF. The performance difference between System 4 and System 5 has vanished on the SdSVC test set. Notably, results on the VoxCeleb1 test set remain strong and often improve after applying domain-balanced HPM, despite the reduced VoxCeleb sampling frequency.

We conduct additional experiments to separately study the impact of the increased sampling frequency of Farsi and the focus on harder samples during training. Results of these experiments can be found in Table 2. We fine-tune the System 5 baseline with the protocol described in Section 2.2, but do not take the speaker similarity into account and just randomly sample imposter speakers from the same domain. One experiment balances the domain of speakers (*balanced*) while another experiment exclusively samples from the in-domain (*Farsi*) training set. In addition, we compare our domain-balanced HPM approach against the broad HPM of Section 2.2.1 and against an HPM variant that only samples from Farsi speakers.

Table 2: *Effects of fine-tuning (FT) and our proposed HPM strategies.*

Method	Domain	Vox1	SdSVC 2020	
		EER(%)	EER(%)	MinDCF
baseline	-	0.87	2.13	0.0938
HPM	Farsi	2.00	2.01	0.0910
HPM	broad	0.83	1.98	0.0875
HPM	balanced	**0.79**	**1.69**	**0.0759**
FT	Farsi	6.05	1.83	0.0854
FT	balanced	0.87	1.82	0.0802

Basic fine-tuning of the systems on SdSVC training data only, increases the in-domain performance significantly with a relative improvement of 14.1% and 9.0% in EER and MinDCF respectively. Balancing the sampling frequency however, prevents the degradation on the VoxCeleb1 test set and further improves the MinDCF by 6.1% relative. The EER remains stable. This indicates that it is worthwhile to keep out-of-domain performance stable while fine-tuning the systems.

The importance of domain-balancing increases when applying our proposed HPM strategy. As the balance between the domain sampling increases, so does the performance on both evaluation sets. Incorporating HPM on top of domain-balanced sampling shows to be beneficial and increases relative performance with 7.1% and 5.4% in EER and MinDCF respectively.

3.3. Language-dependent score normalization

As shown in Table 1, the language-dependent variant of our adaptive s-normalization system further improves EER and MinDCF values on average with 3.3% and 2.3% respectively on the SdSVC test set. While modest, the improvement is consistent and easy applicable in the scoring backend.

To analyze the impact of different imposter speaker cohorts, we analyze the HPM domain-balanced System 5 with different s-norm configurations. The results on the SdSVC test set are provided in Table 3. The imposter cohort is restricted to the top-40 most similar imposters for all experiments.

Table 3: *Effects of cohort selection in s-normalization.*

Imposter Cohort	EER(%)	MinDCF
no s-normalization	2.14	0.0947
VoxCeleb	2.46	0.1303
Farsi	**1.69**	**0.0759**
VoxCeleb + Farsi	1.72	0.0762

The results clearly illustrate that a cohort restricted to the available in-domain training data proves to be the most optimal configuration. We notice a relative improvement of 21% and 19.9% in EER and MinDCF respectively over a system without s-norm.

3.4. Final submission

The final score-based fusion of the single systems fine-tuned with domain-balanced HPM and language-dependent score normalization results in an EER of 1.45% and a MinDCF of 0.0651 as shown in Table 1. Fusion of all systems leads to a relative improvement over System 5 of 11% and 11.9% in EER and MinDCF respectively on the SdSVC test set. This shows that minor architectural variations can prove sufficient to learn complementary speaker embeddings.

4. Conclusion

In this paper we presented HPM as a computationally efficient hard negative mining strategy to fine-tune a speaker embedding extractor towards out-of-domain Farsi data. Furthermore, a correct configuration of s-normalization has proved to be crucial to handle the cross-lingual trials presented in the SdSV Challenge 2020. A fusion of five systems based on our ECAPA-TDNN architecture in conjunction with the proposed techniques resulted in a final top-scoring submission on Task 2 of the SdSVC with an EER of 1.45% and a MinDCF of 0.065.

5. References

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

[2] D. Snyder, D. Garcia-Romero, G. Sell, A. McCree, D. Povey, and S. Khudanpur, "Speaker recognition for multi-speaker conversations using x-vectors," in *Proc. ICASSP*, 2019, pp. 5796–5800.

[3] A. Nagrani, J. S. Chung, and A. Zisserman, "VoxCeleb: A large-scale speaker identification dataset," in *Proc. Interspeech*, 2017, pp. 2616–2620.

[4] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep speaker recognition," in *Proc. Interspeech*, 2018, pp. 1086–1090.

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

[6] 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 *Proc. ASRU*, 2019, pp. 397–402.

[7] H. Zeinali, K. A. Lee, J. Alam, and L. Burget, "Short-duration speaker verification (SdSV) challenge 2020: the challenge evaluation plan," 2019.

[8] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," in *Proc. Interspeech*, 2020.

[9] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation networks," in *Proc. IEEE/CVF CVPR*, 2018, pp. 7132–7141.

[10] S. Gao, M.-M. Cheng, K. Zhao, X. Zhang, M.-H. Yang, and P. H. S. Torr, "Res2Net: A new multi-scale backbone architecture," *IEEE TPAMI*, 2019.

[11] Z. Gao, Y. Song, I. McLoughlin, P. Li, Y. Jiang, and L.-R. Dai, "Improving Aggregation and Loss Function for Better Embedding Learning in End-to-End Speaker Verification System," in *Proc. Interspeech*, 2019, pp. 361–365.

[12] D. Snyder, G. Chen, and D. Povey, "MUSAN: A music, speech, and noise corpus," 2015.

[13] 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 *Proc. ICASSP*, 2017, pp. 5220–5224.

[14] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Proc. Interspeech*, 2019.

[15] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in *2019 IEEE/CVF CVPR*, 2019, pp. 4685–4694.

[16] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Proc. ICLR*, 2014.

[17] L. N. Smith, "Cyclical learning rates for training neural networks," in *IEEE WACV*, 2017, pp. 464–472.

[18] Z. N. Karam, W. M. Campbell, and N. Dehak, "Towards reduced false-alarms using cohorts," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 4512–4515.

[19] S. Cumani, P. Batzu, D. Colibro, C. Vair, P. Laface, and V. Vasilakakis, "Comparison of speaker recognition approaches for real applications," in *Proc. Interspeech*, 2011, pp. 2365–2368.

[20] N. Brümmer and E. de Villiers, "The BOSARIS toolkit: Theory, algorithms and code for surviving the new DCF," 2013.

[21] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," *Proc. IEEE CVPR*, pp. 815–823, 2015.

[22] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. NeurIPS*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., 2017, pp. 4077–4087.

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

[24] 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," 2020.

[25] P. Matejka, O. Novotný, O. Plchot, L. Burget, M. Diez, and J. Černocký, "Analysis of score normalization in multilingual speaker recognition," in *Proc. Interspeech*, 2017, pp. 1567–1571.

[26] M. F. BenZeghiba, J. Gauvain, and L. Lamel, "Gaussian backend design for open-set language detection," in *Proc. ICASSP*, 2009, pp. 4349–4352.