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

Domain-Invariant Speaker Vector Projection by Model-Agnostic Meta-Learning

Jiawen Kang[1], *Ruiqi Liu*[1,2], *Lantian Li*[1*], *Yunqi Cai*[1,3], *Dong Wang*[1*], *Thomas Fang Zheng*[1]

[1]Center for Speech and Language Technologies, Tsinghua University, Beijing
[2]China University of Mining and Technology, Beijing
[3]Department of Computer Science and Technology, Tsinghua University, Beijing

lilt@cslt.org; wangdong99@mails.tsinghua.edu.cn

Abstract

Domain generalization remains a critical problem for speaker recognition, even with the state-of-the-art architectures based on deep neural nets. For example, a model trained on reading speech may largely fail when applied to scenarios of singing or movie. In this paper, we propose a domain-invariant projection to improve the generalizability of speaker vectors. This projection is a simple neural net and is trained following the Model-Agnostic Meta-Learning (MAML) principle, for which the objective is to classify speakers in one domain if it had been updated with speech data in another domain. We tested the proposed method on CNCeleb, a new dataset consisting of single-speaker multi-condition (SSMC) data. The results demonstrated that the MAML-based domain-invariant projection can produce more generalizable speaker vectors, and effectively improve the performance in unseen domains.

Index Terms: speaker recognition, meta-learning, domain generalization

1. Introduction

Speaker recognition has gained good performance after decades of research [1]. Most modern approaches are based on *speech embedding*, i.e., representing variable-length speech segments by fixed-length continuous vectors. This embedding is traditionally derived from statistical models, e.g., the i-vector model [2], and recently mostly via deep neural nets (DNN) [3, 4], e.g., the x-vector model [5, 6]. The deep embedding models have been significantly improved recently, by employing better architectures [7, 8], pooling approaches [6, 9, 10, 11], training objectives [12, 13, 14, 15, 16], and training schemes [17, 18, 19]. As a result, it has achieved the state-of-the-art (SOTA) performance on several benchmark datasets [20], in particular when combined with the PLDA model [21] for scoring.

In spite of the high performance on existing benchmark datasets, a large performance degradation is often observed when the deep embedding models are deployed to real applications. For example, in a preliminary study [22], we found that a SOTA model trained with the large-scale Voxceleb dataset can achieve great performance on the SITW evaluation set (less than 2% in equal error rate (EER)), however when applied to a more realistic CNCeleb evaluation set, the performance degrades to 10%-30% in EER, depending on the genre of the test data. This degradation should be attributed to the severe domain mismatch caused by the complex acoustic environments and speaking styles in real-life applications. Unfortunately, this mismatch is not easy to solve by simply collecting more data, compared to other speech processing tasks such as automatic speech recognition (ASR). This is because the speaker property is convolved with other factors in the speech signal. In order to distinguish the speaker property from other factors, the training data must contain speech from the same speaker but in different acoustic environments and speaking styles, i.e., single-speaker and multi-condition (SSMC) data. In contrast, ASR training requires single-word and multi-condition (SWMC) data. It is obvious that SSMC data is much more difficult to collect than SWMC data.

A large body of research has been conducted to solve the domain mismatch problem. The most popular approach is domain adaptation, which adapts the basic model by a small amount of in-domain data. Since the embedding model is highly complex, the adaptation is more often performed with the PLDA scoring model. This adaptation could be supervised or unsupervised. The supervised approach uses class labels in the target domain, and adapts PLDA following the Bayesian rule in principle [23, 24]. The unsupervised approach employs various clustering methods to generate pseudo classes, and then treats these pseudo classes as true speakers to conduct supervised adaptation [25, 26]. Another approach is domain-invariant training. Compared to the adaptation approach that targets for better performance in a particular domain, the domain-invariant training targets for learning domain-insensitive speaker vectors, and is more amiable to real applications where the conditions may vary in time. For example, Wang et al. [27] proposed an adversarial loss that prevents the produced speaker vectors from being domain discriminative.

In this paper, we present a domain-invariant training approach based on meta learning. Meta learning is a high-level learning strategy with the principle of knowledge sharing and transferring among tasks [28, 29]. In the deep learning regime, early meta-learning approaches learn a training scheme [30, 31]. Recently, Finn et al. presented a new Model-Agnostic Meta-Learning (MAML) algorithm [32], which employs data of multiple tasks to learn a model that can be easily adapted to a new task. Borrowing this idea, we propose a **robust MAML** algorithm that can learn domain-invariant model directly, rather than a model that is ready for adaptation as in the standard MAML. We apply this new algorithm to learn a projection net that improves the domain invariance of the raw deep speaker vectors (x-vectors in our experiments). Experimental results on a new SSMC CNCeleb dataset [22] demonstrated that this approach is promising.

The rest of the paper is organized as follows. Section 2 reviews the basic MAML algorithm, and Section 3 presents the details of the robust MAML algorithm and the MAML-based project net. Section 4 presents the experimental result, and Section 5 gives a conclusion of the entire paper.

[*] Corresponding authors

http://dx.doi.org/10.21437/Interspeech.2020-2562

2. MAML algorithm

Meta learning is a long-standing theme in machine learning [28, 29]. The central idea of this learning approach is to reuse the knowledge of some *prior* tasks to speed up the learning of a new task. The knowledge can be either the learned models or the learning strategy (how to learn a task). For neural models, reusing both types of knowledge is easy. In the former case, it is known as *multi-task learning* or *transfer learning* [33], and in the latter case, a more proper name is *learning to learn* [30, 31].

Model-Agnostic Meta-Learning (MAML) [32] is a new meta learning approach. The concept is shown in Figure 1(a). In this picture, there are two prior tasks $\mathcal{T}_1$ and $\mathcal{T}_2$, and the associated training and test data are (T_1^r, T_1^t) and (T_2^r, T_2^t), respectively. Let f_θ denotes the function of the model parameterized by θ. For each mini-batch, a small set of training data from $\mathcal{T}_i$, denoted by $m_i^r \sim T_i^r$, is selected. Using this mini-batch, the gradient ∇_θ is computed, which is then used to perform a *local update* for the model:

$$\theta' = \theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta; m_i^r), \tag{1}$$

where $\mathcal{L}_{\mathcal{T}_i}$ is the loss function of task $\mathcal{T}_i$, and α is the learning rate. Based on the new parameters θ', compute the loss on $m_i^t \sim T_i^t$, a mini-batch from the test dataset of the *same* task:

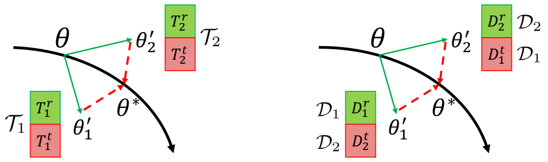
$$\mathcal{L}_{\mathcal{T}_i}(f_{\theta'}; m_i^t) = \mathcal{L}_{\mathcal{T}_i}(f_{\theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta; m_i^r)}; m_i^t). \tag{2}$$

This loss is used to compute gradient for model update, which is called the *meta update*.

$$\theta \leftarrow \theta - \beta \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_{\theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta; m_i^r)}; m_i^t), \tag{3}$$

where β is the learning rate. It should be noted that it is the meta update that truly modifies the model parameters. The local update is just a proxy to compute the gradient for the meta update.

From the training procedure, it can be seen that the goal of MAML is to find an optimal θ^* at which *the averaged performance would be best if a simple gradient update had been conducted to adapt the present model to task-specific models*. In other words, MAML intends to learn a good initial model base on which task-specific models can be easily obtained by one or a few gradient updates.



(a) MAML

(b) Robust MAML

Figure 1: *MAML for training a good initial model (a) and robust MAML for training a domain-invariant model (b). The green solid line (green block) represents local update, and the red dash line (red block) represents the meta update.*

3. MAML-based domain-invariant projection

3.1. Robust MAML

The MAML algorithm discussed in the previous section focuses on a good initial model. This is valuable for many applications especially those with limited training data. It can be directly applied to speaker recognition for domain adaptation. Specifically, we can treat the speaker recognition task on each domain as a particular prior task in the MAML algorithm, and train an initial model with data from a couple of prior domains. When deploying to a new domain, a small amount of training data would be sufficient to adapt the initial model to a domain-specific model. However, this adaptation approach does not meet our original goal of designing a robust model that works well on *any unseen* domain. We therefore present a robust MAML to deal with the problem.

As shown in Figure 1(b), we have two domains $\mathcal{D}_1$ and $\mathcal{D}_2$, and $\{D_1^r, D_1^t\}$ and $\{D_2^r, D_2^t\}$ are the training and test sets for each of the two domains, respectively. The MAML training is conducted as usual, but the meta update is based on a mini-batch whose domain is different from the one used for the local update. Put it formally, the local update is conducted by randomly choosing a mini-batch in the i-th domain:

$$\theta' = \theta - \alpha \nabla_\theta \mathcal{L}(f_\theta; m_i^r), \tag{4}$$

where $m_i^r \sim D_i^r$. Note that we have omitted the domain dependency in the loss function $\mathcal{L}$ as all the domains share the same loss function. During the meta update, a mini-batch from the j-th domain is selected, and the loss function is as follows:

$$\mathcal{L}(f_{\theta'}; m_j^t) = \mathcal{L}(f_{\theta - \alpha \nabla_\theta \mathcal{L}(f_\theta; m_i^r)}; m_j^t), \tag{5}$$

where $m_j^t \in D_j^t$, and usually m_i^r and m_i^t are from different domains. The meta update is then formulated as follows:

$$\theta \leftarrow \theta - \beta \nabla_\theta \mathcal{L}(f_{\theta - \alpha \nabla_\theta \mathcal{L}(f_\theta; m_i^r)}; m_j^t). \tag{6}$$

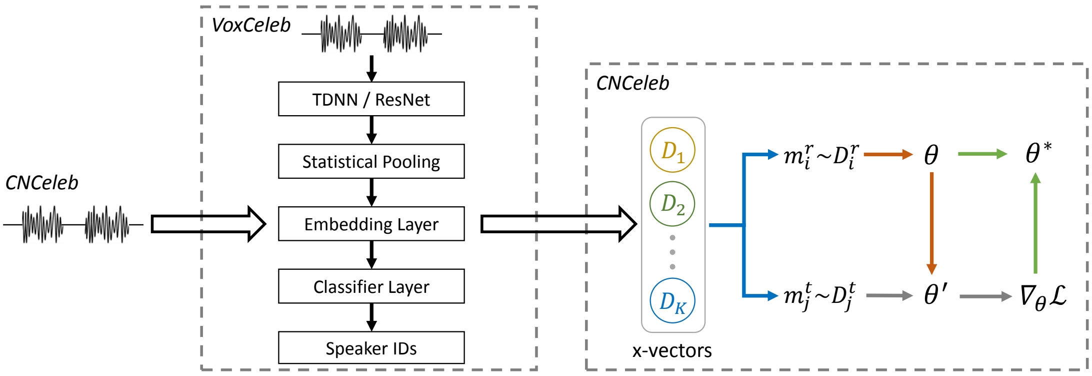
Choosing different domains for the local update and the meta update is important. Assume that the model has converged to θ^*, it is easy to see that updating θ^* towards *any* domain will result in good performance for *all* domains. This implies that θ^* is a stationary point that works well for all the prior domains even without any update. The idea that training conducted in one domain and evaluated in other domains aligns to robust optimization [34]. We therefore denote the new algorithm as **robust MAML**. Note that a similar idea has been discussed in [35].

We highlight that with the robust MAML, it is not necessary that the local update and the meta update use the same set of speakers, although better performance was found if they do. It means that SSMC data is not strictly required for MAML training. This is a key advantage compared to other domain-robust approaches, for example multi-conditional training.

3.2. Domain-invariant projection net

At the first glance, applying the robust MAML to train domain-invariant speaker embedding models is straightforward. However, we found it is not applicable in real situations. A particular problem is domain imbalance: for most of the existing datasets, a large proportion of the data are clean and reading speech, and only a small amount of data are from other domains. With this imbalanced data, the minor domains will be overwhelmed by the major domain when conducting the robust MAML training.

To solve this problem, we choose a post-processing scheme. Firstly, we use a standard large-scale dataset to train the main embedding model, which is the x-vector model in our experiment. Secondly, we apply the robust MAML algorithm to train an extra projection net that maps the original x-vectors to a new vector space where domain invariance is improved. Figure 2



(a) Deep speaker embedding (b) Robust MAML

Figure 2: *Robust MAML-based domain-invariant projection on deep speaker vectors.*

illustrates the entire training procedure, where K denotes the number of domains.

4. Experiments

4.1. Data

Three datasets were used in our experiments: VoxCeleb [7, 36], SITW [37] and CNCeleb [22]. More information about these three datasets is presented below.

VoxCeleb: This is a large-scale audiovisual speaker database collected by the University of Oxford. The entire database contains $2,000+$ hours of speech signals from $7,000+$ speakers. It was used to train the x-vector embedding model and the basic LDA/PLDA scoring model. Data augmentation was applied to improve robustness, with the MUSAN corpus used to generate noisy utterances, and the room impulse responses (RIRS) corpus used to generate reverberant utterances.

SITW: This is a standard evaluation dataset excerpted from VoxCeleb1. In our experiments, the Eval.Core test set, which contains $3,658$ target trials and $718,130$ imposter trials, was used for evaluation.

CNCeleb: This is a large-scale free speaker recognition dataset collected by Tsinghua University. It contains more than 130k utterances from $1,000$ Chinese celebrities. It covers 11 diverse domains, and each speaker may have speech samples in multiple domains, therefore is a true SSMC dataset [22]. The entire dataset was split into two parts: *CNCeleb.Train*, which covers 7 domains including entertainment, play, vlog, live broadcast, speech, drama and recitation, and involves $45,370$ utterances from 800 speakers, was used to train the projection net and the LDA/PLDA scoring model. *CNCeleb.Eval*, which covers the rest 3 domains including singing, movie and interview, and involves $8,729$ utterances from 200 speakers, was used for evaluation in unseen domains.

4.2. Embedding models

We built two x-vector embedding models, one is based on TDNN, and the other is based on ResNet. Both are widely used in speaker recognition research.

TDNN: This was trained using the Kaldi toolkit [38], following the SITW recipe. The acoustic features are 40-dimensional FBanks. The main architecture contains three components. The first component involves 5 time-delay (TD) layers to learn frame-level speaker features. The second component computes the mean and standard deviation of the frame-level features. The third component involves 2 full-connection (FC) layers and outputs the posterior probability over the $7,185$ speakers of the VoxCeleb dataset. Once trained, the 512-dimensional activations of the penultimate FC layer are read out as the x-vector of the input utterance.

ResNet34: The ResNet architecture is similar to the TDNN architecture, with two differences: (1) it uses the ResNet-34 structure to learn frame-level speaker features [39]; (2) it uses the Additive Angular Marginal Softmax (AAM-Softmax) [40] to compute the posterior probabilities over the training speakers. Again, the 512-dimensional activations of the penultimate FC layer are read out as the x-vector of the input utterance.

4.3. Projection networks

The projection network is designed to transform the raw x-vectors from the embedding model to a new vector space where domain invariance is improved.

4.3.1. MAML net

In our experiments, the projection net involves 3 FC layers, and every layer consists of 512 units. The loss function is the same as the one used for the embedding model, which is the standard Softmax for the TDNN model, and the AAM-Softmax for the ResNet34 model. The projection net can be trained with CNCeleb.Train, by using the robust MAML algorithm. We denote the projection net trained in this way as a *MAML net*. Once the training is completed, the domain-invariant x-vectors can be obtained from the penultimate FC layer.

4.3.2. MCT net

Using the same architecture and the same loss function as the MAML net, we can train the projection net using the regular training scheme. Since the training data (CNCeleb.Train) is SSMC, this is essentially a multi-conditional training (MCT). We call the MCT-trained projection net as a *MCT net*. Similar to the MAML net, the MCT net can improve the domain invariance of speaker vectors. However, the MCT net purely relies on the SSMC data, while the MAML net relies both the SSMC data and the robust training scheme.

4.4. Baseline results

We firstly test the performance of the baseline systems, i.e., both the embedding model and the LDA/PLDA scoring model are trained with VoxCeleb, without any additional speaker vector projection. We test the performance on the SITW Eval.Core test set and also the CNCeleb.Eval test sets, including three individual domains (singing, movie and interview). The results in terms of equal error rate (EER) are reported in Table 1, where the results with two scoring methods, cosine scoring and LDA/PLDA scoring, are reported, respectively. For the LDA/PLDA scoring, the x-vectors are firstly pre-processed by LDA, and then are used to compute scores by PLDA. The dimensionality of the LDA projection was set to 128 in our experiments.

Table 1: *Performance (EER%) of the baseline systems.*

Test Set	TDNN		ResNet34	
	Cosine	LDA/PLDA	Cosine	LDA/PLDA
SITW.Eval.Core	5.139	2.433	3.226	1.968
CNC.Eval.Singing	29.95	26.88	28.47	27.18
CNC.Eval.Movie	26.09	20.24	25.19	21.29
CNC.Eval.Interview	19.68	15.97	19.23	15.47

4.5. Results with domain-invariant projection

In this experiment, we test the MAML net and MCT net on the CNCeleb.Eval test sets, where the domains are never seen in the training data of the embedding models, the LDA/PLDA scoring models and the projection networks.

4.5.1. Performance with cosine scoring

First look at the performance with the cosine scoring. Since the complex back-end scoring models are not used, we can evaluate the true quality of the speaker vectors. The results on CNCeleb.Eval test sets are presented in Table 2. For convenience, the baseline results are presented in the 'Base' columns.

It can be observed that in all the test conditions, both the MCT net and MAML net can substantially improved the system performance, and the MAML net offers more significant improvement. This demonstrates that the multi-conditional training scheme with SSMC data is an effective way to improve domain invariance (MCT vs. Ori), and the robust MAML training scheme can provide additional and substantial contribution (MAML vs. MCT).

Table 2: *Performance (EER%) with cosine scoring.*

Cosine	TDNN			ResNet34		
Domain	Base	MCT	MAML	Base	MCT	MAML
Singing	29.95	30.85	29.86	28.47	28.40	**27.08**
Movie	26.09	25.46	24.27	25.19	24.92	**24.21**
Interview	19.68	17.51	**16.82**	19.23	16.92	16.87

To give a better comparison between the MCT scheme and the MAML scheme, Figure 3 shows their performance along with the training process. It can be seen that for both the MCT net and the MAML net, the EER continuously reduces. Moreover, the MAML net delivers better performance than the MCT net. Compared the two ResNet34 curves in both pictures, we observe that the MCT net seems overfitting after 4k iterations, while the MAML net is generally healthy.

4.5.2. Performance with LDA/PLDA scoring

We finally test the performance with LDA/PLDA scoring. In this experiments, all the LDA/PLDA scoring models were trained with CNCeleb.Train. For the baseline system, we retrained the LDA/PLDA models with the raw x-vectors from the embedding model. For the MCT and MAML systems, the LDA/PLDA models were trained with the x-vectors produced by the MCT net and MAML net. Again, the dimensionality of the LDA projection was set to 128. Since the training data (CNCeleb.Train) is SSMC, this training scheme of LDA/PLDA is essentially a multi-conditional training (MCT), and therefore should be robust against domain variance in a way.

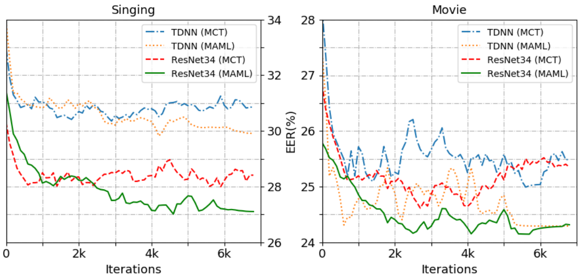


Figure 3: *Performance (EER%) on two unseen domains during the training process of MCT net and MAML net.*

The results are shown in Table 3. Firstly, compared to the results in Table 1, it can be observed that the MCT-based LDA/PLDA models offer dramatic performance improvement on the test data. Secondly, the contributions of the MCT net and the MAML net are both marginal. This is not surprising, as the x-vectors for LDA/PLDA training and MCT/MAML nets training are duplicated. From the perspective of practitioners, this is a good thing as it implies that domain invariance could be largely attained by retraining the LDA/PLDA scoring model with SSMC data. On the other hand, it suggests that the MAML training scheme should be extended to the scoring model, otherwise its contribution on the speaker vectors will be diminished.

Table 3: *Performance (EER%) with LDA/PLDA scoring.*

LDA/PLDA	TDNN			ResNet34		
Domain	Base	MCT	MAML	Base	MCT	MAML
Singing	25.67	25.50	25.35	23.83	23.66	**23.53**
Movie	19.63	18.74	18.85	18.19	**17.75**	**17.75**
Interview	13.63	13.47	13.58	12.05	**11.85**	**11.85**

5. Conclusions

This paper proposed a domain-invariant projection to improve the generalizability of speaker vectors. We presented a robust MAML algorithm to train the projection net, which promotes domain invariance not only by the SSMC data, but also by the robust training scheme. Experimental results on the CNCeleb dataset demonstrated that the speaker vectors produced by MAML-based projection are more domain-invariant compared to the raw x-vectors and the speaker vectors produced by a multi-conditional trained projection. This leads to significant performance improvement with cosine scoring. However, when the scoring model is an LDA/PLDA that was trained with SSMC data, the contribution of the projection net seems marginal. Future work will investigate the MAML-based training for the LDA/PLDA scoring model, and investigate the light-weighted MAML-based adaptation.

6. Acknowledgements

This work was supported by the National Natural Science Foundation of China No. 61633013. Jiawen Kang and Ruiqi Liu are joint first authors.

7. References

[1] J. H. Hansen and T. Hasan, "Speaker recognition by machines and humans: A tutorial review," *IEEE Signal processing magazine*, vol. 32, no. 6, pp. 74–99, 2015.

[2] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. 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] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *ICASSP*, 2014, pp. 4052–4056.

[4] L. Li, Y. Chen, Y. Shi, Z. Tang, and D. Wang, "Deep speaker feature learning for text-independent speaker verification," in *INTERSPEECH*, 2017, pp. 1542–1546.

[5] 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.

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

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

[8] J. weon Jung, H.-S. Heo, J. ho Kim, H. jin Shim, and H.-J. Yu, "RawNet: Advanced end-to-end deep neural network using raw waveforms for text-independent speaker verification," in *INTERSPEECH*, 2019, pp. 1268–1272.

[9] W. Cai, J. Chen, and M. Li, "Exploring the encoding layer and loss function in end-to-end speaker and language recognition system," in *Proceedings of Odyssey: The Speaker and Language Recognition Workshop*, 2018, pp. 74–81.

[10] W. Xie, A. Nagrani, J. S. Chung, and A. Zisserman, "Utterance-level aggregation for speaker recognition in the wild," in *ICASSP*, 2019, pp. 5791–579.

[11] N. Chen, J. Villalba, and N. Dehak, "Tied mixture of factor analyzers layer to combine frame level representations in neural speaker embeddings," in *INTERSPEECH*, 2019, pp. 2948–2952.

[12] W. Ding and L. He, "MTGAN: Speaker verification through multitasking triplet generative adversarial networks," in *INTERSPEECH*, 2018, pp. 3633–3637.

[13] J. Wang, K.-C. Wang, M. T. Law, F. Rudzicz, and M. Brudno1, "Centroid-based deep metric learning for speaker recognition," in *ICASSP*, 2019, pp. 3652–3656.

[14] Z. Bai, X.-L. Zhang, and J. Chen, "Partial AUC optimization based deep speaker embeddings with class-center learning for text-independent speaker verification," *arXiv preprint arXiv:1911.08077*, 2019.

[15] 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 *INTERSPEECH*, 2019, pp. 361–365.

[16] J. Zhou, T. Jiang, Z. Li, L. Li, and Q. Hong, "Deep speaker embedding extraction with channel-wise feature responses and additive supervision softmax loss function," in *INTERSPEECH*, 2019, pp. 2883–2887.

[17] R. Li, N. L. D. Tuo, M. Yu, D. Su, and D. Yu, "Boundary discriminative large margin cosine loss for text-independent speaker verification," in *ICASSP*, 2019, pp. 6321–6325.

[18] S. Wang, J. Rohdin, L. Burget, O. Plchot, Y. Qian, K. Yu, and J. Cernocky, "On the usage of phonetic information for text-independent speaker embedding extraction," in *INTERSPEECH*, 2019, pp. 1148–1152.

[19] T. Stafylakis, J. Rohdin, O. Plchot, P. Mizera, and L. Burget, "Self-supervised speaker embeddings," in *INTERSPEECH*, 2019, pp. 2863–2867.

[20] S. O. Sadjadi, C. Greenberg, E. Singer, D. Reynolds, L. Mason, and J. Hernandez-Cordero, "The 2018 NIST speaker recognition evaluation," in *INTERSPEECH*, 2019, pp. 1483–1487.

[21] S. Ioffe, "Probabilistic linear discriminant analysis," in *European Conference on Computer Vision (ECCV)*. Springer, 2006, pp. 531–542.

[22] Y. Fan, J. Kang, L. Li, K. Li, H. Chen, S. Cheng, P. Zhang, Z. Zhou, Y. Cai, and D. Wang, "CN-CELEB: a challenging chinese speaker recognition dataset," in *ICASSP*, 2020.

[23] J. Villalba and E. Lleida, "Bayesian adaptation of PLDA based speaker recognition to domains with scarce development data," in *Proceedings of Odyssey: The Speaker and Language Recognition Workshop*, 2012, pp. 47–54.

[24] D. Garcia-Romero and A. McCree, "Supervised domain adaptation for i-vector based speaker recognition," in *ICASSP*, 2014, pp. 4047–4051.

[25] D. Garcia-Romero, A. McCree, S. Shum, and C. Vaquero, "Unsupervised domain adaptation for i-vector speaker recognition," in *Proceedings of Odyssey: The Speaker and Language Recognition Workshop*, 2014.

[26] B. Sun, J. Feng, and K. Saenko, "Return of frustratingly easy domain adaptation," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[27] Q. Wang, W. Rao, S. Sun, L. Xie, E. S. Chng, and H. Li, "Unsupervised domain adaptation via domain adversarial training for speaker recognition," in *ICASSP*, 2018, pp. 4889–4893.

[28] R. Vilalta and Y. Drissi, "A perspective view and survey of meta-learning," *Artificial intelligence review*, vol. 18, no. 2, pp. 77–95, 2002.

[29] J. Vanschoren, "Meta-learning: A survey," *arXiv preprint arXiv:1810.03548*, 2018.

[30] S. Bengio, Y. Bengio, J. Cloutier, and J. Gecsei, "On the optimization of a synaptic learning rule," in *Preprints Conf. Optimality in Artificial and Biological Neural Networks*, vol. 2. Univ. of Texas, 1992.

[31] M. Andrychowicz, M. Denil, S. Gomez, M. W. Hoffman, D. Pfau, T. Schaul, B. Shillingford, and N. De Freitas, "Learning to learn by gradient descent by gradient descent," in *Advances in neural information processing systems*, 2016, pp. 3981–3989.

[32] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 1126–1135.

[33] D. Wang and T. F. Zheng, "Transfer learning for speech and language processing," in *APSIPA ASC*, 2015, pp. 1225–1237.

[34] Q. Qian, S. Zhu, J. Tang, R. Jin, B. Sun, and H. Li, "Robust optimization over multiple domains," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 4739–4746.

[35] Q. Dou, D. C. de Castro, K. Kamnitsas, and B. Glocker, "Domain generalization via model-agnostic learning of semantic features," in *Advances in Neural Information Processing Systems*, 2019, pp. 6447–6458.

[36] A. Nagrani, J. S. Chung, and A. Zisserman, "VoxCeleb: a large-scale speaker identification dataset," in *INTERSPEECH*, 2017.

[37] M. McLaren, L. Ferrer, D. Castan, and A. Lawson, "The speakers in the wild (SITW) speaker recognition database." in *INTERSPEECH*, 2016, pp. 818–822.

[38] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The Kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, 2011.

[39] H. Zeinali, S. Wang, A. Silnova, P. Matějka, and O. Plchot, "BUT system description to voxceleb speaker recognition challenge 2019," *arXiv preprint arXiv:1910.12592*, 2019.

[40] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4690–4699.