Reducing Domain Mismatch by Maximum Mean Discrepancy Based Autoencoders

Weiwei LIN, Man-Wai MAK, and Longxin LI

Dept. of Electronic and Information Engineering,
The Hong Kong Polytechnic University,
Hong Kong SAR

Jen-Tzung Chien

Dept. of Electrical and Computer Engineering,
National Chiao Tung University,
Taiwan

Abstract

Domain mismatch, caused by the discrepancy between training and test data, can severely degrade the performance of speaker verification (SV) systems. What's more, both training and test data themselves could be composed of heterogeneous subsets, with each subset corresponding to one sub-domain. These multi-source mismatches can further degrade SV performance. This paper proposes incorporating maximum mean discrepancy (MMD) into the loss function of autoencoders to reduce theses mismatches. Specifically, we generalize MMD to measure the discrepancies among multiple distributions. We call this generalized MMD domain-wise MMD. Using domain-wise MMD as an objective function, we derive a domain-invariant autoencoder (DAE) for multi-source i-vector adaptation. The DAE directly encodes the features that minimize the multi-source mismatch. By replacing the original i-vectors with these domain-invariant feature vectors for PLDA training, we reduce the EER by 11.8% in NIST 2016 SRE when compared to PLDA without adaptation.

1. Introduction

Using i-vector as an unsupervised feature extraction method and PLDA as a supervised channel compensation technique have been very successful in speaker verification [1, 2]. However, like many machine learning algorithms, i-vector/PLDA assumes that the training data and test data are independently sampled from the same distribution. When training data and test data have a severe mismatch, the performance degrades rapidly [3–9]. Mismatch between training data and test data is not uncommon, as it can be caused by a lot of factors such as languages, channels, noises, and genders. Basically, collecting more data to retrain the system is time-consuming and computationally-expensive; such a solution is also unrealistic in some scenarios. It is desirable to use the existing data and a small amount of target-specific data to modify the system to meet the need, which is essentially what domain adaptation (DA) does.

Early attempts in i-vector based DA require the in-domain data to have speaker labels. For example, Garcia-Romero and McCree [3] computed the MAP-estimates of the in-domain within-speaker and across-speaker covariance matrices in the i-vector space using the speaker labels from the in-domain data. In [5], these matrices are treated as latent variables and their joint posterior distribution is factorized using variational Bayes so that the MAP point estimates of the matrices can be computed from the factorized distributions. The point estimates are then used for scoring in the in-domain environment. Another approach is to generate *hypothesized* speaker labels via unsupervised clustering [4, 10, 11]. Given the hypothesized labels, the covariance matrices of in-domain data can be computed as usual and can be interpolated with the out-of-domain covariance matrices to obtain an adapted PLDA model. Of course, correctly inferring all of the missing labels is even harder than performing speaker verification. However, as shown in [4], even imperfect labels can achieve performance almost as good as the correct labels. Still, cluster-based approaches require a lot of heuristics to set the number of clusters.

It is also possible to carry out the unsupervised DA without inferring the missing labels at all. Most of the methods in this category assume that there is a common feature space in which the in-domain and the out-domain have a minimum mismatch. DA aims to project data onto such feature space and uses the projected data to train a classifer. As mismatch can be caused by multiple sources, it is helpful to divide the training data into subsets according to their sources before finding a common feature space. This is called multi-source domain adaptation in the literature [12]. In addition to the robustness to heterogeneous sources, this approach also has the potential to generalize to unseen domains, as it does not assume a particular in-domain environment. The inter-dataset variability compensation (IDVC) [6] is a typical example of this approach. IDVC divides the training data into several subsets, and for each subset, the mean is computed. The means of these subsets are used to find the directions of maximum inter-dataset variability; then the subspace corresponding to these directions is removed from all i-vectors.

Several theoretical works in DA [13–15] suggest that minimizing the divergence between the in-domain and out-domain distributions is very important for obtaining a good representation for DA. From this perspective, approaches based solely on the differences among the domain-means, such as IDVC, are not enough for finding a good representation. The reason is that even if the means of the distributions are exactly the same, there could still be severe mismatch between the data distributions if their variances are very different. Thus, to reduce inter-dataset mismatch, it is important to consider the statistics beyond the means.

To better utilize the statistics of multi-source data, we consider using maximum mean discrepancy (MMD) as an objective function for measuring multi-source mismatches. Maximum mean discrepancy is a nonparametric method for measuring the distance between two distributions [16–18]. With a properly chosen kernel, MMD can utilize all moments of data. We gen-

This work was in part supported by the RGC of Hong Kong SAR with Grant No. PolyU 152518/16E and Taiwan MOST with Grant 107-2634-F-009-003.

eralize MMD to measure the discrepancies among multiple distributions. Then, we use the generalized MMD as an objective function for training autoencoders. With this objective function, the autoencoders learn the features that contain less domain-specific information but are still relevant to the classification task. Because the ultimate goal of the autoencoders is to make the feature vectors invariant to domain mismatch, we refer to them as **d**omain-invariant **a**uto**e**ncoders (DAE).[1]

2. The I-vector/PLDA Framework

Since its first appearance [1], i-vector has become the de facto choice for the representation of utterances in speaker verification and other related areas. The i-vector approach is essentially a factor analysis (FA) technique trying to find a low-dimensional subspace that captures most of the variations in the GMM-supervectors. Specifically, the GMM-supervector [20] of utterance t can be generated by the following generative model:

$$\boldsymbol{\mu}_t = \boldsymbol{\mu} + \mathbf{T}\mathbf{w}_t, \tag{1}$$

where $\boldsymbol{\mu}$ is a supervector formed by stacking the means of a universal background model (UBM) and $\mathbf{T}$ is a low-rank total variability matrix. The posterior mean of $\mathbf{w}_t$ is the i-vector $\mathbf{x}_t$ of utterance t.

As i-vector contain all sort of variabilities in utterances, channel compensation techniques are essential for suppressing the non-speaker variability. Among them, probabilistic discriminant analysis (PLDA) [2] performs the best. Given a set of D-dimensional length-normalized [21] i-vectors $\{\mathbf{x}_{ij}; i = 1, \ldots, N; i = 1, \ldots, H_i;\}$ from N speakers, each with H_i sessions, PLDA assumes that the i-vectors can be expressed as the following factor analysis model:

$$\mathbf{x}_{ij} = \mathbf{m} + \mathbf{V}\mathbf{z}_i + \boldsymbol{\epsilon}_{ij}, \tag{2}$$

where $\mathbf{m}$ is the global mean of the i-vectors, $\mathbf{V}$ defines the speaker subspace, $\mathbf{z}_i$ is the speaker factor and $\boldsymbol{\epsilon}_{ij}$ is the residual noise.

3. Maximum Mean Discrepancy Autoencoder

In this section, we first highlight the domain mismatches in NIST 2016 SRE data and the limitation of IDVC. Then, we explain why maximum mean discrepancy is theoretically better than IDVC and how it can be incorporated into the training of autoencoders for extracting domain-invariant features.

3.1. Multi-source Mismatch in NIST 2016 SRE

NIST 2016 speaker recognition evaluation (SRE16) introduces various new challenges to speaker recognition [22, 23], among which the multilingual setup brought the most attention. Unlike previous SREs, both development (Dev) and evaluation (Eval) data in SRE16 comprise utterances spoken in non-English languages. Table 1 shows the composition of SRE16 data. Because all of the SRE16 data are non-English, training using data from previous SREs results in poor performance. Training using only SRE16 data is also not feasible, as there are only 2,472 segments in total and a very small number of them are labeled. Besides, the labeled development data have different languages than the evaluation data.

[1] Do not confuse with the denoising autoencoder [19].

Dataset	Category	Language
Dev	Unlabeled	Cantonese and Tagalog
Dev	Unlabeled	Mandarin and Cebuano
Dev	Labeled	Mandarin and Cebuano
Eval	Enrolment	Cantonese and Tagalog
Eval	Test	Cantonese and Tagalog

Table 1: The composition of SRE16 data. "Labeled" means speaker labels are provided. "unLabeled" means speaker labels are not provided.

Fig. 1 shows the t-distributed stochastic neighbor embedding (t-SNE) [24] of i-vectors from SRE16 development data and previous SRE data. In the figure, datasets are colored according to their genders and languages. We can see that there are significant mismatches in terms of cluster means and cluster variances. Also, the multi-source mismatches occur not only between the English data (ENG_F and ENG_M) and SRE16 data but also within SRE16 data (CAN_F, CAN_M, TGL_F and TGL_M).

3.2. Inter-dataset Variability Compensation

Inter-dataset variability compensation (IDVC) was proposed in [6]. IDVC follows the subspace removal approach proposed in [25]. It aims to find the directions in the i-vector space with the largest inter-dataset variability and removes the i-vector variability in these directions. This is achieved by projecting the i-vectors $\mathbf{x}$'s as follows:

$$\hat{\mathbf{x}} = (\mathbf{I} - \mathbf{W}\mathbf{W}^{\mathsf{T}})\mathbf{x}, \tag{3}$$

where the columns of $\mathbf{W}$ span the subspace of unwanted variability. $\mathbf{W}$ comprises of the eigenvectors of the covariance matrix of the subset means. Note that in IDVC the domain mismatch is defined by the variances and covariances of subset means.

However, the mismatch of datasets may not only manifest in the dataset means, but also in the higher-order statistics of these datasets. The limitation of IDVC will become apparent when we consider some Gaussian distributions (one for each dataset) with identical means but different covariance matrices. Despite of the severe mismatches among these Gaussians, IDVC considers these Gaussians to be identical and will not remove any subspace ($\mathbf{W}$ in Eq. 3 is a null matrix) to reduce the mismatches.

3.3. Maximum Mean Discrepancy

The theoretical works in DA [13–15] suggest that it is important to have a good measurement of the divergence between the data distributions of different domains. Maximum mean discrepancy is a distance measure on the space of probability. Given two sets of samples $\{\mathbf{x}_i\}_{i=1}^N$ and $\{\mathbf{y}_j\}_{j=1}^M$, MMD computes the mean squared difference of the statistics of the two datasets:

$$\mathcal{D}_{\text{MMD}} = \left\| \frac{1}{N}\sum_{i=1}^{N}\phi(\mathbf{x}_i) - \frac{1}{M}\sum_{j=1}^{M}\phi(\mathbf{y}_j) \right\|^2, \tag{4}$$

where ϕ is a feature map. When ϕ is the identity function, the MMD distance simply computes the discrepancy between the

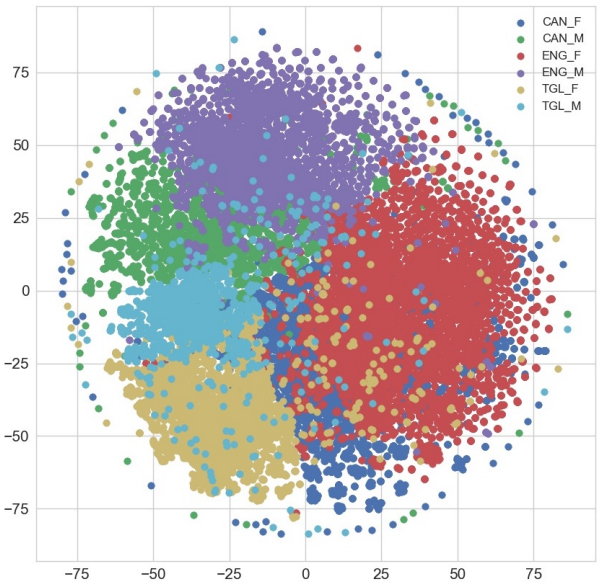

Figure 1: Scatter plot of 2-dimensional t-SNE embedded i-vectors. In the legend, "M" and "F" stand for male and female, respectively, and "CAN", "ENG" and "TGL" stand for Cantonese, English and Tagalog, respectively.

sample means. Eq. 4 can be expanded as:

$$\mathcal{D}_{\text{MMD}} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{i'=1}^{N}\phi(\mathbf{x}_i)^{\mathsf{T}}\phi(\mathbf{x}_{i'}) - \frac{2}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}\phi(\mathbf{x}_i)^{\mathsf{T}}\phi(\mathbf{y}_j) + \frac{1}{M^2}\sum_{j=1}^{M}\sum_{j'=1}^{M}\phi(\mathbf{y}_j)^{\mathsf{T}}\phi(\mathbf{y}_{j'}). \tag{5}$$

As each term in Eq. 5 involves dot products only, the kernel trick can be applied:

$$\mathcal{D}_{\text{MMD}} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{i'=1}^{N}k(\mathbf{x}_i,\mathbf{x}_{i'}) - \frac{2}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}k(\mathbf{x}_i,\mathbf{y}_j) + \frac{1}{M^2}\sum_{j=1}^{M}\sum_{j'=1}^{M}k(\mathbf{y}_j,\mathbf{y}_{j'}), \tag{6}$$

where $k(\cdot,\cdot)$ is a kernel function.

3.4. MMD as Autoencoder's Loss Function

Assume that we have in-domain data $\{\mathbf{x}_i^{\text{in}}\}_{i=1}^{N_{\text{in}}}$ and out-domain data $\{\mathbf{x}_j^{\text{out}}\}_{j=1}^{N_{\text{out}}}$. We want to learn a transform $\mathbf{h} = f(\mathbf{x})$ such that the transformed data $\{\mathbf{h}_i^{\text{in}}\}_{i=1}^{N_{\text{in}}}$ and $\{\mathbf{h}_j^{\text{out}}\}_{j=1}^{N_{\text{out}}}$ are as similar as possible. The mismatch between the transformed data can be measured by MMD:

$$\mathcal{D}_{\text{MMD}} = \frac{1}{N_{\text{in}}^2}\sum_{i=1}^{N_{\text{in}}}\sum_{i'=1}^{N_{\text{in}}}k(\mathbf{h}_i^{\text{in}},\mathbf{h}_{i'}^{\text{in}}) - \frac{2}{N_{\text{in}}N_{\text{out}}}\sum_{i=1}^{N_{\text{in}}}\sum_{j=1}^{N_{\text{out}}}k(\mathbf{h}_i^{\text{in}},\mathbf{h}_j^{\text{out}}) + \frac{1}{N_{\text{out}}^2}\sum_{j=1}^{N_{\text{out}}}\sum_{j'=1}^{N_{\text{out}}}k(\mathbf{h}_j^{\text{out}},\mathbf{h}_{j'}^{\text{out}}). \tag{7}$$

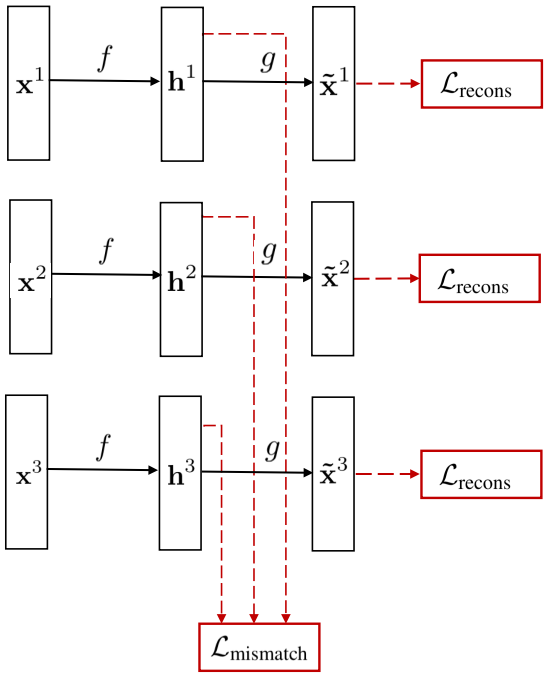

Figure 2: Architecture of the proposed domain-invariant autoencoder (DAE) when data are from three different domains. Solid black arrows represent the connections between neurons. Dashed red arrows represent the hidden nodes' outputs for computing the domain-mismatch loss or autoencoder's outputs for computing the reconstruction loss.

When the data come from multiple sources, we want the transformed data to be as similar to each other as possible. To this end, we define a domain-wise MMD measure. Specifically, given D sets of data $\{\mathbf{x}_i^d\}_{i=1}^{N_d}$, where $d = 1, 2, \ldots, D$, we want the transformed data $\{\mathbf{h}_i^d\}_{i=1}^{N_d}$ to have small loss as defined by the following equation:

$$\mathcal{L}_{\text{mismatch}} = \sum_{d=1}^{D}\sum_{\substack{d'=1\\d'\neq d}}^{D}\left(\frac{1}{N_d^2}\sum_{i=1}^{N_d}\sum_{i'=1}^{N_d}k(\mathbf{h}_i^d,\mathbf{h}_{i'}^d) - \frac{2}{N_dN_{d'}}\sum_{i=1}^{N_d}\sum_{j=1}^{N_{d'}}k(\mathbf{h}_i^d,\mathbf{h}_j^{d'}) + \frac{1}{N_{d'}^2}\sum_{j=1}^{N_{d'}}\sum_{j'=1}^{N_{d'}}k(\mathbf{h}_j^{d'},\mathbf{h}_{j'}^{d'})\right). \tag{8}$$

Of course, we also want to retain as much non-domain related information as possible. Assume that another transform can reconstruct the input from $\mathbf{h}$:

$$\tilde{\mathbf{x}} = g(\mathbf{h}), \tag{9}$$

where $\tilde{\mathbf{x}}$ is the reconstruction of the input $\mathbf{x}$. We want to make $\tilde{\mathbf{x}}$ as close to $\mathbf{x}$ as possible by minimizing:

$$\mathcal{L}_{\text{recons}} = \frac{1}{2}\sum_{d=1}^{D}\sum_{i=1}^{N_d}\left\|\mathbf{x}_i^d - \tilde{\mathbf{x}}_i^d\right\|^2. \tag{10}$$

Both objectives can be achieved by an antoencoder comprising an encoder network f and a decoder network g, with the loss

function:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{mismatch}} + \lambda \mathcal{L}_{\text{recons}}, \quad (11)$$

where λ is a parameter controlling the importance of the reconstruction loss. Note that both f and g can be multilayer neural networks. As the autoencoder encodes domain-invariant information, we call it domain-invariant autoencoder (DAE). Fig. 2 shows the architecture of a DAE for three domains ($D = 3$), with each row corresponding to one domain. Note that the weights in the rows are shared across all domains.

4. Experiment Setup

4.1. Speech Data and Acoustic Features

Speech files from NIST 2004–2010 Speaker Recognition Evaluation (hereafter, referred to as SRE04–SRE10)[2] and the development set of SRE16 (SRE16-dev) were used as development data and speech files from the evaluation set of SRE16 (SRE16-eval) were used as test data. The speech regions in the speech files were extracted by using a two-channel voice activity detector [26]. For each speech frame, 19 MFCCs together with energy plus their 1st and 2nd derivatives were computed, followed by cepstral mean normalization and feature warping [27] with a window size of three seconds. A 60-dim acoustic vector was extracted every 10ms, using a Hamming window of 25ms.

4.2. I-vector Extraction and PLDA Model Training

I-vectors derived from SRE04–SRE10 were used for training a DAE (Fig. 2) with 300 hidden nodes and IDVC's projection matrix ($\mathbf{W}$ in Eq. 3) with rank $= 6$. The resulting networks and projection matrix were then applied to the i-vectors derived from SRE16. Then, principal component analysis (PCA) was applied to the adapted i-vectors to reduce the dimension to 200. Within-class covariance normalization (WCCN) and i-vector length normalization were applied to the 200-dimensional i-vectors [21, 28]. Then, we trained a gender-independent PLDA model with 200 latent variables. PLDA scores were normalized by S-norm using SRE16 development data as the cohort set [29].

4.3. MMD Autoencoders and IDVC Training

The weights in the encoder and decoder networks of the DAE are tied as in [30]. The DAE were trained by minimizing $\mathcal{L}_{\text{total}}$ in Eq. 11 using the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L–BFGS) algorithm [31]. To train the IDVC project matrix and the DAE, we divided SRE04–10 and SRE16 data into gender and language dependent subsets, each corresponding to one domain. Excluding the minor data in SRE16, we have six subsets: English male, English female, Cantonese male, Cantonese female, Tagalog male and Tagalog female.

5. Results and discussions

5.1. General Performance Analysis

Table 2 shows the performance of IDVC and DAE. All systems use PLDA as their backend. A classical i-vector PLDA system without domain adaptation (No Adapt) is also included for comparison. For the DAE, we used a quadratic kernel

$$k(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^{\mathsf{T}}\mathbf{y} + 1)^2$$

and set λ in Eq. 11 to 1.0. We used one hidden layer with 300 hidden units. The network has the structure 300(input)–300(hidden)–300(output). For a more in-depth discussion of the hype-parameter λ and the kernel choice, readers may refer to [**?**].

We can see from Table 2 that both DAE and IDVC boost the performance significantly in term of EER, although in terms of minimum Cprimary and actual Cprimary, the improvement is minor. We can also observe that DAE has a small improvement over IDVC in terms of EER.

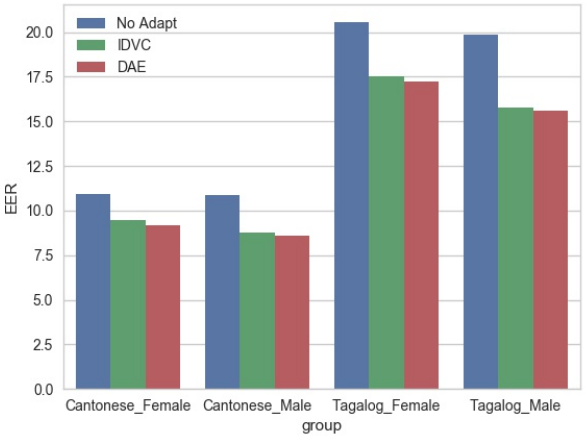

Figure 3: EER performance of IDVC and DAE in the gender- and language-dependent subsets of SRE16-eval.

5.2. Subset Performance Analysis

To gain more insights into the performance of the DA methods, we report the performance of the three systems on four gender- and language-dependent subsets in Table 3 and Fig 3. The results suggest that Tagalog is more challenging than Cantonese, with 20.55% and 19.89% EER in the male and the female, respectively. Also, the female subsets seem to be more difficult than the male ones. The performance of the four subsets improves significantly after both domain adaptation methods. The DAE has small improvement over IDVC on all of the four subsets.

	EER(%)	mCprim	aCprim
No Adapt	15.84	0.89	0.93
IDVC	13.08	0.86	0.93
DAE	12.79	0.85	0.91

Table 2: The Performance of DAE and IDVC and the performance of a classical i-vector PLDA system without domain adaptation in the SRE16 evaluation set. The DAE uses linear activations in the hidden nodes. "mCprim" and "aCprim" are the minimum detection cost and the actual detection cost as specified in the evaluation plan of SRE16.

[2] https://www.nist.gov/itl/iad/mig/speaker-recognition

	Cantonese						Tagalog					
	Female			Male			Female			Male		
	EER(%)	Cprim	ACprim	EER(%)	Cprim	ACprim	EER(%)	Cprim	ACprim	EER(%)	Cprim	ACprim
No Adapt	10.92	0.77	0.87	10.87	0.74	0.96	20.55	0.93	0.94	19.89	0.94	0.96
IDVC	9.47	0.74	0.88	8.74	0.68	0.96	17.50	0.91	0.93	15.75	0.90	0.96
DAE	9.15	0.73	0.84	8.61	0.67	0.94	17.26	0.90	0.91	15.59	0.89	0.94

Table 3: The performances of IDVC and DAE on the subsets of the SRE16 evaluation set.

6. Conclusions and Future Work

In this paper, we proposed a domain-invariant autoencoder (DAE) for multiple-source i-vector domain adaptation. Unlike IDVC, with a quadratic kernel, the DAE can utilize the first and the second moments of data for measuring the domain mismatch. The experiments on SRE16 show that the DAE can significantly improve SV performance. The experiments also demonstrate that the proposed methods have small improvement over IDVC.

7. References

[1] Najim Dehak, Patrick 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.

[2] Peng Li, Yun Fu, U. Mohammed, J.H. Elder, and S.J.D Prince, "Probabilistic models for inference about identity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 144–157, 2012.

[3] Daniel Garcia-Romero and Alan McCree, "Supervised domain adaptation for i-vector based speaker recognition," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 4047–4051.

[4] Daniel Garcia-Romero, Alan McCree, Stephen Shum, Niko Brummer, and Carlos Vaquero, "Unsupervised domain adaptation for i-vector speaker recognition," in *Proc. Odyssey*, 2014, pp. 260–264.

[5] Jesús Villalba and Eduardo Lleida, "Bayesian adaptation of PLDA based speaker recognition to domains with scarce development data," in *Proc. Odyssey*, 2012, pp. 47–54.

[6] Hagai Aronowitz, "Inter dataset variability compensation for speaker recognition," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 4002–4006.

[7] Mitchell McLaren and David Van Leeuwen, "Source-normalized LDA for robust speaker recognition using i-vectors from multiple speech sources," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 3, pp. 755–766, 2012.

[8] Ondrej Glembek, Jeff Ma, Pavel Matejka, Bing Zhang, Oldrich Plchot, Lukas Burget, and Spyros Matsoukas, "Domain adaptation via within-class covariance correction in i-vector based speaker recognition systems," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2014, pp. 4032–4036.

[9] Alexey Sholokhov, Tomi Kinnunen, and Sandro Cumani, "Discriminative multi-domain PLDA for speaker verification," in *Proc. IEEE International Conference onAcoustics, Speech and Signal Processing*. IEEE, 2016, pp. 5030–5034.

[10] Stephen H. Shum, Douglas A. Reynolds, Daniel Garcia-Romero, and Alan McCree, "Unsupervised clustering approaches for domain adaptation in speaker recognition systems," in *Proc. Odyssey*, 2014, pp. 265–272.

[11] Longxin Li and Man-Wai Mak, "Unsupervised domain adaptation for gender-aware PLDA mixture models," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018.

[12] Gabriela Csurka, "Domain adaptation for visual applications: A comprehensive survey," *arXiv preprint arXiv:1702.05374*, 2017.

[13] Shai Ben David, Tyler Lu, Teresa Luu, and Dávid Pál, "Impossibility theorems for domain adaptation," in *Proc. the Thirteenth International Conference on Artificial Intelligence and Statistics*, 2010, pp. 129–136.

[14] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh, "Domain adaptation: Learning bounds and algorithms," *arXiv preprint arXiv:0902.3430*, 2009.

[15] Pascal Germain, Amaury Habrard, François Laviolette, and Emilie Morvant, "A PAC-Bayesian approach for domain adaptation with specialization to linear classifiers," in *Proc. International Conference on Machine Learning*, 2013, pp. 738–746.

[16] Arthur Gretton, Karsten M Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex J. Smola, "A kernel method for the two-sample-problem," in *Proc. Advances in Neural Information Processing systems*, 2007, pp. 513–520.

[17] Yujia Li, Kevin Swersky, and Rich Zemel, "Generative moment matching networks," in *Proc. International Conference on Machine Learning*, 2015, pp. 1718–1727.

[18] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan, "Learning transferable features with deep adaptation networks," in *Proc. International Conference on Machine Learning*, 2015, pp. 97–105.

[19] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

[20] William M. Campbell, Douglas E. Sturim, and Douglas A. Reynolds, "Support vector machines using gmm supervectors for speaker verification," *IEEE signal processing letters*, vol. 13, no. 5, pp. 308–311, 2006.

[21] Daniel Garcia-Romero and Carol Y Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Proc. Interspeech*, 2011, pp. 249–252.

[22] Seyed Omid Sadjadi, Timothe Kheyrkhah, Audrey Tong, Craig Greenberg, Douglas Reynolds, Elliot Singer, Lisa Mason, and Jaime Hernandez-Cordero, "The 2016 NIST speaker recognition evaluation," in *Proc. Interspeech*, 2017, pp. 1353–1357.

[23] Karen Jones, Stephanie Strassel, Kevin Walker, David Graff, and Jonathan Wright, "Call My Net corpus: A multilingual corpus for evaluation of speaker recognition technology," in *Proc. Interspeech 2017*, 2017, pp. 2621–2624.

[24] Laurens van der Maaten and Geoffrey Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.

[25] Alex Solomonoff, Carl Quillen, and William M. Campbell, "Channel compensation for SVM speaker recognition.," in *Proc. Odyssey*, 2004, vol. 4, pp. 219–226.

[26] Man-Wai Mak and Hon-Bill Yu, "A study of voice activity detection techniques for NIST speaker recognition evaluations," *Computer Speech & Language*, vol. 28, no. 1, pp. 295–313, 2014.

[27] Jason Pelecanos and Sridha Sridharan, "Feature warping for robust speaker verification," in *Proc. Odyssey*, 2001.

[28] Andrew O Hatch, Sachin S Kajarekar, and Andreas Stolcke, "Within-class covariance normalization for SVM-based speaker recognition.," in *Proc. Interspeech 2016*, 2006.

[29] Pavel Matejka, Ondrej Novotnỳ, Oldrich Plchot, Lukáš Burget, and JH Cernockỳ, "Analysis of score normalization in multilingual speaker recognition," in *Proc. Interspeech 2017*, 2017, pp. 1348–1352.

[30] Geoffrey E. Hinton and Ruslan R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[31] Dong C. Liu and Jorge Nocedal, "On the limited memory BFGS method for large scale optimization," *Mathematical Programming*, vol. 45, no. 1, pp. 503–528, 1989.