Speech Communication 129 (2021) 7–16

Contents lists available at ScienceDirect

Speech Communication

journal homepage: www.elsevier.com/locate/specom

An investigation of domain adaptation in speaker embedding space for speaker recognition

Fahimeh Bahmaninezhad, Chunlei Zhang, John H.L. Hansen *

Center for Robust Speech Systems (CRSS), Erik Jonsson School of Engineering & Computer Science, The University of Texas at Dallas, Richardson, TX, USA

ARTICLE INFO

Keywords:
Speaker recognition
Domain adaptation
Unsupervised adaptation
NIST SRE-16 and SRE-18
Discriminant analysis
Triplet loss function

ABSTRACT

Speaker recognition continues to grow as a research challenge in the field with expanded application in commercial, forensic, educational and general speech technology interfaces. However, challenges remain, especially for naturalistic audio streams including recordings with mismatch between train and test data (i.e., when train or system development data and enrollment/test data or application data are collected from different sources). Mismatch conditions (Hansen and Hasan, 2015) can be divided into two categories, extrinsic (channel, noise, etc.) and intrinsic (duration, language, and speaker traits including stress, emotion, Lombard effect, vocal effort, accent). Here, we investigate speaker recognition for the domain mismatch problem (intrinsic mismatch) especially for those challenges introduced by NIST (National Institute of Standards and Technology) SRE (speaker recognition evaluation) in 2016 and 2018. The challenges introduced in NIST SRE-16 and SRE-18 include language mismatch between train (used for the development of the system) and enrollment/test (used at the application phase). Here, we develop three alternative speaker embedding systems; i-vector, t-vector (an improved triplet loss solution), and x-vector. In addition, a number of unsupervised and supervised (using pseudo labels) methods are also studied for domain mismatch compensation, especially applied at the back-end level. These include adapted PLDA, adapted discriminant analysis, as well as score normalization and calibration methods using unlabeled in-domain data. We propose new variations to discriminant analysis with support vectors (SVDA) as well. These results confirm that SVDA can measurably improve speaker recognition performance for SRE-16 and SRE-18 tasks respectively by +15% and +8% in terms of min-Cprimary; and for EER the gains are +14% and +16% respectively, using i-vector speaker embeddings as the baseline. These advancements offer promising steps toward addressing speaker recognition in naturalistic audio streams.

1. Introduction

Speaker recognition (SR) is defined as recognizing whether a specific target speaker is talking during a given speech segment or not (Sadjadi et al.; Hansen and Hasan, 2015). Here, we focus on the text-independent SR task which does not require any constraint on the speech content. Approaches proposed for SR have evolved significantly over the past few years to overcome the limitations and variations of training/in-domain data as well as to provide consistent performance on naturalistic audio streams (Hansen and Hasan, 2015); however, challenges still remain, especially for intrinsic mismatch conditions.

Over the previous two decades, NIST (National Institute of Standards and Technology) has been organizing SRE (speaker recognition evaluation) tasks regularly to encourage participating sites to focus on the specific problem of making SR systems robust against realistic data and core technology issues (Sadjadi et al.). NIST SRE tasks have covered a wide range of challenges targeting different training and test conditions to make speaker recognition systems effective enough for conversational telephone speech (with limited training data). Also, providing robust solutions for combination of speaker detection and recognition tasks under various mismatch conditions (including channel variations, duration mismatch, and language mismatch).

Challenges organized for speaker recognition evaluation (including NIST SRE) had played an important role to lead researchers to migrate from GMM (Gaussian mixture model)-UBM (universal background model) (Reynolds et al., 2000) based systems toward i-Vector and deep learning based solutions (i.e., t-Vector (Zhang et al., 2018), x-Vector (Snyder et al.), and etc (Bahmaninezhad and Hansen, 2018)). During this migration, we can also include methods based on: joint factor analysis (JFA) (Kenny et al., 2007), i-Vector (Dehak et al.,

* Corresponding author.
E-mail addresses: fahimeh.bahmaninezhad@utdallas.edu (F. Bahmaninezhad), chunlei.zhang@utdallas.edu (C. Zhang), john.hansen@utdallas.edu (J.H.L. Hansen).

https://doi.org/10.1016/j.specom.2021.01.001
Received 1 December 2019; Received in revised form 5 May 2020; Accepted 7 January 2021
Available online 23 January 2021

2011) solutions with cosine distance scoring or support vector machine (SVM) classification (Dehak et al., 2011), and i-vector with PLDA (probabilistic linear discriminant analysis) scoring (Ioffe, 2006; Dehak et al., 2011) as well. These also include both UBM/i-Vector and DNN/i-Vector, where i-Vector/PLDA had been the state-of-the-art method for speaker recognition (depending on the data) as well as other speech areas, such as language recognition. These advancements in SR had provided a satisfactory performance on NIST SRE tasks until 2012. However, for challenges introduced in the SRE-16 and SRE-18, current solutions are not sufficiently effective and require further investigation. More specifically, the NIST SRE-16 and SRE-18 focused on the domain mismatch problem (training data used for development had different language sets than those in enrollment/test data which are used at the application phase; there were handset and microphone mismatch options as well).

Domain mismatch compensation for speaker recognition has been previously studied for diverse datasets and tasks (other than SRE-16 and SRE-18), including (Misra and Hansen, 2018; Garcia-Romero et al., 2014; Misra and Hansen, 2014; Shum et al., 2014; Shon et al., 2017; Bahmaninezhad and Hansen, 2016). For NIST SRE tasks specifically, multiple studies have proposed methods to compensate for domain mismatch. Here, we consider several earlier works on these tasks. Generally speaking, domain mismatch compensation techniques can be applied to speaker recognition systems at different phases: *front-end* level compensation (e.g., MAP – maximum a posterior – adaptation of GMMs model (Colibro et al., 2017), speaker embedding extraction), and *back-end* level (e.g., PLDA adaptation (Snyder et al.)). Fig. 1 represents an overall block-diagram of an i-vector based speaker recognition system specifying front-end and back-end level processing. From an alternative viewpoint, domain mismatch compensation methods can be categorized into *supervised* or *unsupervised* techniques as well. When in-domain data are unlabeled, pseudo labeling can be integrated into the system to provide for supervised adaptation.

To compensate for the domain mismatch at the speaker-embedding extraction level (i.e., front-end), (Colibro et al., 2017) introduced GMM-SVM with Nuisance Attribute Projection (NAP) trained using clustered unlabeled in-domain data for SRE-16 task. They also studied other methods for unsupervised domain mismatch compensation, using in-domain data for MAP adaptation of GMM models which both were shown to be effective. In addition, Plchot et al. (2017) proposed training a speaker classifier neural network for the extraction of d-vectors. Interestingly, they did not attempt to assign pseudo speaker labels to the unlabeled data. Borgstrom et al. (2017) applied an unsupervised Bayesian adaptation method and achieved promising results. Snyder et al. (2017) replaced i-vectors with two new proposed embeddings which are derived based on a DNN architecture. They evaluated the performance of these embeddings on both SRE-10 and SRE-16 tasks, although the idea is general and not necessarily developed for domain adaptation, experiments show that the discriminative training of speaker embeddings can help toward domain mismatch compensation rather than traditional i-vector embeddings. x-vector (Snyder et al.) which uses data augmentation as well as PLDA adaptation is among the top-performing systems for SRE-16 and SRE-18 (where a small unlabeled in-domain data is provided for the adaptation purpose) tasks.

Overall, the performance of most of these methods has been reported along with other modifications at the back-end level. For x-Vector, the back-end level techniques are shown to directly impact the superiority of x-Vector over the i-Vector. Therefore, it would be difficult to draw a conclusion on how effective front-end level domain adaptation might be, considering the fact that available in-domain data for SRE tasks is very limited.

To compensate for domain mismatch at the dimension reduction step, Plchot et al. (2017) used LDA with a within-class covariance correction (WCC) technique, which updates the within-class covariance matrix using in-domain data. Mismatch compensation at the score calculation and score normalization steps have also been studied in Colibro et al. (2017), where they added replicate copies of in-domain data to the training set for modeling the classifiers. In addition, Colibro et al. (2017) used the in-domain data in multiple score normalization techniques. Torres-Carrasquillo et al. (2017) not only applied whitening and mean centralization using in-domain data (both labeled and unlabeled) but also proposed a multi-stage PLDA adaptation technique (which uses clustered unlabeled data). They also incorporated in-domain data into the score normalization step as well. Plchot et al. (2017) normalized the resulting scores using speaker-dependent s-norm with a cohort created from training and unlabeled in-domain data. Two studies Lee et al. (2017), Torres-Carrasquillo et al. (2017) also mentioned the use of unlabeled data for score calibration. These techniques have all been proposed to compensate for domain mismatch at the back-end level and have been shown to be effective for NIST SRE tasks. Other studies on the SRE tasks that target domain mismatch compensation include (Madikeri et al., 2016; Rouvier et al., 2016; Lee et al., 2019; Bhattacharya et al., 2019).

Successful submission at NIST SRE challenges requires fusion of multiple complementary systems. Most of the studies on NIST SRE tasks have reported a combination of effective modules that work together. In some cases, the papers may only report scores on the DEV set or EVAL set; however, one important issue can be on the generalization from DEV set toward the EVAL set which has been missed in some reports. In this study, we present a comprehensive study on back-end level domain mismatch compensation techniques with emphasizing the contribution of each of them separately and in combination with other techniques. In addition, we report both DEV and EVAL scores for throughout and comprehensive comparison of different compensation techniques and their generalization capabilities.

Based on our experience for SRE-16 (Zhang et al., 2017) and SRE-18 (Zhang et al., 2019) tasks, we realized domain adaptation (especially unsupervised domain adaptation) is a key strategy in achieving acceptable submission, which is our main focus here. In this study, we introduce and analyze our proposed solutions for compensating the domain mismatch problem. We specifically perform domain adaptation at the back-end level, as (i) in-domain data is very limited to be applied in the speaker embedding extraction phase, (ii) unlabeled in-domain data can be risky for integration into the front-end level processing. Solutions introduced in this study benefit from:

- mean centralization of data with unlabeled in-domain data;
- training LDA/PLDA while limiting the training data to only those which benefit back-end modeling;
- using SVDA (support vector discriminant analysis) (Bahmaninezhad and Hansen, 2017) to alleviate the domain mismatch problem;
- unlabeled in-domain data clustered to be used with LDA/PLDA;
- supervised (using pseudo labels) and unsupervised adaptation of PLDA;
- using unlabeled data for score calibration and fusion.

Although we provide more detailed explanation and analysis of our proposed solutions throughout this paper, the main contributions of this study can be summarized as (1) evaluation of i-Vector, x-Vector, and t-Vector solutions for both SRE-16 and SRE-18 task, (2) proposed supervised and unsupervised SVDA, (3) supervised and unsupervised PLDA adaptation with different speaker embeddings, (4) score normalization, calibration and fusion with effective utilization of unlabeled data, and (5) comprehensive evaluation of each technique for both DEV and EVAL sets which studies their effectiveness alone, or in combination, with other compensation methods. We evaluate the performance of our systems on both SRE-16 and SRE-18 tasks, which target both language mismatch problem with additional unlabeled data provided for system development.

In Section 2, we first introduce our speaker embedding systems developed and evaluated in this study. Section 3 defines the NIST

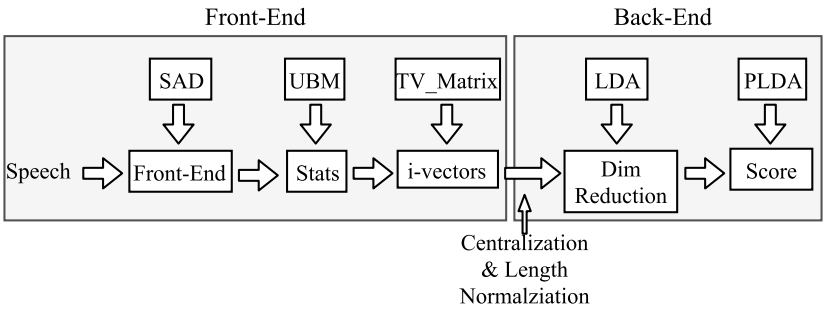

Fig. 1. Block-diagram of i-vector/PLDA speaker recognition.

SRE-16 and NIST SRE-18 tasks specifically. Next, in Section 4, we formulate our solutions for compensation of domain mismatch and their performance is evaluated in Section 5. Finally, conclusions and discussions are provided in Section 6.

2. Speaker recognition

Speaker recognition refers to the task of recognizing whether a target speaker is talking during a given test segment or not. Here, we develop three different speaker recognition systems based on a traditional UBM/i-Vector (Dehak et al., 2011; Zhang et al., 2017), t-Vector (an improved triplet loss solution, where an additional L_2 constrained softmax loss term is introduced to formulate a multi-task learning objective based speaker embedding system (Zhang and Koishida, 2017a,b; Zhang et al., 2018; Li et al., 2017)), and x-vector (Povey et al., 2011; Snyder et al., 2017; Snyder et al.) frameworks. Each of these systems are described in detail in the following subsections.

2.1. UBM/i-vector system

Historically, i-Vector based systems achieve great success not only in speaker recognition (Dehak et al., 2011; Zhang et al., 2017; Lee et al., 2017; Sadjadi et al.), but also in language recognition (Bulut et al., 2017). The block diagram of our baseline i-Vector/PLDA speaker recognition is shown in Fig. 1. In the i-Vector framework, a channel and speaker-dependent GMM supervector is factorized as,

$$M = m + Tw, \tag{1}$$

where m is the UBM speaker and channel-dependent supervector, and T is the low rank total variability matrix (TV-matrix) which maps the high-dimensional GMM supervector into w, known as i-vector.

i-Vectors are next post-processed using mean centralization, length normalization (Garcia-Romero and Espy-Wilson, 2011), and LDA (linear discriminant analysis). Scoring is performed with PLDA (Ioffe, 2006) (the back-end processing is the same for the other two speaker embeddings as well). Finally, given two i-vectors $\hat{w}_1$ and $\hat{w}_2$ at the recognition phase, we need to determine whether these two belong to the same speaker (target) or not (non-target) with the following log-likelihood ratio,

$$\text{log-likelihood} = log\frac{p(\hat{w}_1, \hat{w}_2|\text{target})}{p(\hat{w}_1, \hat{w}_2|\text{non-target})}. \tag{2}$$

2.2. x-Vector system

The x-Vector has been reported to achieve very effective speaker recognition performance in recent studies (Snyder et al., 2017; Snyder et al.). The model is a deep neural network (DNN) based speaker discriminative framework benefiting from practical techniques such as data augmentation and statistical pooling. The embeddings are extracted over the entire utterance instead of at the frame-level. The network is trained with a softmax loss function and corresponding speaker labels, given by:

$$\mathcal{L}_s = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{W_{y_i}^T f(\mathbf{x}_i)+b_{y_i}}}{\sum_{j=1}^{C} e^{W_j^T f(\mathbf{x}_i)+b_j}}, \tag{3}$$

where N is the batch size, C is the total speaker number in the training set, $f(\mathbf{x}_i)$ is the output of the embedding layer of the network (i.e., speaker embedding). Here, y_i is the corresponding class label, and W and b are the weights and bias for the last softmax layer of the network which acts as a classifier.

2.3. t-Vector system

Triplet loss is another popular objective function for training face or speaker verification systems (Schroff et al., 2015; Zhang and Koishida, 2017a). The t-Vector system developed here is a modified solution from Zhang et al. (2018), with changes in the loss function and the employed acoustic features. The Inception-resnet-v1 network (Szegedy et al., 2017) (same as Zhang et al. (2018)) is employed here for speaker discriminative training.

Inspired by the success of the softmax loss used in x-Vector models, we perform a modification at the loss function level for the triplet loss based system. Specifically, we formulate a multi-task learning framework by adding an L_2 normalized softmax loss ($\mathcal{L}_{s_{L_2}}$), which is an upgrade of the original softmax loss:

$$\mathcal{L}_{s_{L_2}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{W_{y_i}^T f(\mathbf{x}_i)+b_{y_i}}}{\sum_{j=1}^{C} e^{W_j^T f(\mathbf{x}_i)+b_j}}, \tag{4}$$

subject to $\|f(\mathbf{x}_i)\|_2 = \alpha, \quad \forall i = 1, 2, \ldots, N$

where a simple L_2 normalization is applied to the embedding layer before softmax layer, α is a constant that constrains the radius of the speaker embedding hypersphere. Finally, α is set to 24 empirically in our experiments. With this operation, we are able to better match acoustic conditions between the training and test process (i.e., a L_2-norm embedding layer for softmax training, and the same layer for embedding extraction). The total loss function is an integration of three components: a triplet loss term $\mathcal{L}_{triplet}$, a L_2-norm softmax loss term $\mathcal{L}_{s_{L_2}}$, and a regularization term $\mathcal{L}_2$ which alleviates the over-fitting issue during training,

$$\mathcal{L}_{total} = \mathcal{L}_{triplet} + \omega_1\mathcal{L}_{s_{L_2}} + \omega_2\mathcal{L}_2. \tag{5}$$

Practically, we found the $\omega_1 = 0.1$ and $\omega_2 = 2e-5$ result in a good overall combination for our experiments.

With the update in the loss function, one necessary change is also made in the triplet sampling module. Previously in Zhang et al. (2018), we chose a subset of speakers in the training pool for triplet formulation in each epoch. With the additional $\mathcal{L}_{s_{L_2}}$, it is better to see all the speakers in one epoch. In our experiments, we always randomly select segments from all training speakers for the triplet generation and shuffle to ensure all classes can be seen within one epoch.

Table 1
Statistics of data used in SRE-16 and SRE-18, with model/segment numbers for Enrollment and target/nontarget numbers for Trial.

Unlabeled		Enrollment		Trial	
SRE-16					
Minor	Major	DEV	EVAL	DEV	EVAL
200	2272	80/120	802/1202	4828/19312	1986729/1949666
SRE-18					
		DEV	EVAL	DEV	EVAL
2332		125/175	940/1316	7830/100265	60675/2002332

3. NIST SRE tasks

For developing our systems and evaluation of our proposed methods for the purpose of domain mismatch compensation, we carried out experiments based on both NIST SRE-16 and SRE-18 (Sadjadi et al.; NIST, 2016, 2018). Both target language mismatch problem for speaker recognition where a small unlabeled in-domain data is provided for domain adaptation.

There are two training conditions defined for the NIST SRE (16 and 18 specifically) tasks, (1) fixed: using a fixed dataset for training; (2) open: additional publicly available data are permitted to be used. For all of our experiments, we only focus on the fixed condition.

3.1. NIST SRE-16

NIST SRE-16 fixed condition includes data from Call My Net corpus, previous Mixer/SRE data, both landline and cellular Sadjadi et al.. Here, we did not use the Fisher data and Call-My-Net corpus for training, and at the back-end, we also did not use any of the Switchboard data.

Data assigned to the development and evaluation sets were collected from the Call-My-Net corpus. Data was collected outside of North America and consists of two subsets: (1) *Major*: contains Tagalog and Cantonese languages, (2) *Minor*: contains Cebuano and Mandarin languages. Development data includes data from both minor and major language sets; evaluation data only contains data from the major set (Sadjadi et al.).

Development data includes labeled and unlabeled sets. The labeled set is only from minor languages; 10 speakers talking Cebuano and 10 speakers talking Mandarin, with each possessing 10 segments. The unlabeled set has 2272 and 200 calls from major and minor languages, respectively (this data does not have speaker ID, language, gender, etc information) (Sadjadi et al.). Statistics for both the development and evaluation sets are summarized in Table 1. Throughout the remainder of this paper, we refer to the development set as DEV and evaluation set as EVAL.

3.2. NIST SRE-18

The NIST SRE-18 as well targets a similar challenge with some modifications. For the fixed condition, the training data includes all previous SRE data, consisting of Switchboard, Fisher, VoxCeleb, SITW (speaker in the wild); and the development set of SRE-16 was allowed to be used. The task includes two separate parts: CMN2 (Call My Net), and VAST (Video Annotation for Speech Technology), where for our study here we mainly focus on the CMN2 part. The CMN2 dataset used for the development and evaluation purposes contains data with the Tunisian Arabic language; while the training data is mostly in American English.

In contrast to SRE-16 where DEV and EVAL sets did not share the same languages, SRE-18 DEV and EVAL are considered to belong to one domain (i.e., the language for both are the same). The CMN2 part of the DEV set includes 25 speakers (with approximately 10 utterances per speaker). The SRE-18 DEV set also includes in-domain unlabeled data (no speaker ID, gender, or language labels) with 2332 utterances and speech duration ranging between 10 s to 60 s uniformly.

4. Domain adaptation using unlabeled data

In this section, we review techniques applied specifically at the back-end level for the purpose of domain mismatch compensation. The next section includes experimental results for validating each of these adaptation methods.

4.1. i/t/x-Vector centralization

Mean centralization of the speaker embeddings with in-domain data is shown to be an effective approach for domain adaptation. For SRE-16, we used the major data, and for SRE-18 all unlabeled data are used for centralization processing.

4.2. LDA

The block-diagram in Fig. 1 shows that after extracting i-Vectors (as an example), mean-centralization and length-normalization, usually LDA is used to reduce the dimension size of the resulting i-vectors as well as improve the discriminating ability of the speaker classes. We examine that incorporating in-domain data into the LDA processing can be effective for all i-Vectors, t-Vectors, and x-Vectors. However, LDA is a supervised approach, and adding in-domain data for training the LDA requires some form of pseudo labels for them, which we estimated with the method described in the following subsection.

4.2.1. Clustering of unlabeled data

For compensating the domain mismatch, the use of unlabeled in-domain data becomes very important. There are several stages where we can use unlabeled data, such as LDA/PLDA training, and calibration; however, most of them require labeled data. Therefore, performing a speaker clustering of the unlabeled data is required there. After clustering unlabeled data, we can simply use the "estimated" speaker labels for each utterance with supervised methods. The clustering approach we applied here is similar to the method that we used in our 2015 NIST LRE i-Vector challenge (Bahmaninezhad and Hansen, 2018). With these labels, we incorporate the in-domain information from unlabeled data to train both LDA and PLDA. In fact, in the experiments, this simple operation improves the LDA/PLDA baseline performance for the development set. In practice, we train a gender identification sub-task using previous SRE data before speaker clustering, and then apply a simple K-means algorithm over the gender-dependent subsets, finally, we pool these two subsets together. Throughout our experiments, we found that this can provide more accurate speaker clustering and greater benefits for subsequent LDA and PLDA training.

4.3. Dimension reduction and domain adaptation with SVDA

In this sub-section, we describe discriminant analysis via support vectors (Bahmaninezhad and Hansen, 2017). Here, we modify the SVDA framework for adaptation to the domain of interest.

SVDA is a variation of LDA, where both can be used for discriminant analysis, and optimize the Fisher criterion (Fisher, 1936). LDA uses all samples of all classes to calculate the between and within class covariance matrices, as:

$$S_b = \sum_{c=1}^{C} n_c(\mu_c - \mu)(\mu_c - \mu)^T \tag{6}$$

$$S_w = \sum_{c=1}^{C} \sum_{k \in c} (x_k - \mu_c)(x_k - \mu_c)^T, \tag{7}$$

However, SVDA only uses the support vectors to calculate the between and within class covariance matrices. More specifically, if we define $w_{c_1c_2} = \sum_{i=1}^{l} y_i \alpha_i x_i$ as the optimal direction to classify two classes c_1 and c_2 by a linear SVM (y_i represents target value (+1 for first class,

-1 for second class) of learning pattern x_i, α_i is its coefficient), then the between class covariance matrix will be updated as,

$$V_b = \sum_{1 \le c_1 \le c_2 \le C} w_{c_1 c_2} w_{c_1 c_2}^T. \tag{8}$$

Also, let $\hat{X} = [\hat{x}_1, \hat{x}_2, \dots, \hat{x}_{\hat{N}}]$ be all the support vectors and $\hat{N}$ represents their total number. Next, the within class covariance matrix will be formulated as,

$$V_w = \sum_{c=1}^{C} \sum_{i \in \hat{I}_c} (\hat{x}_i - \hat{\mu}_c)(\hat{x}_i - \hat{\mu}_c)^T \tag{9}$$

where $\hat{I}_c$ includes the index of support vectors in class c, and $\hat{\mu}_c$ denotes the mean of them. Finally, similar to LDA, the optimum transformation $\hat{A}$ will contain the k eigenvectors corresponding to the k largest eigenvalues of $V_w^{-1} V_b$.

For training the SVM, two strategies can be adopted; (i.e., 1-VS-1 and 1-VS-Rest (Bahmaninezhad and Hansen, 2017)). Data for the domain of interest can be easily integrated into this framework, both supervised and unsupervised. In the supervised adapted SVDA, first, the in-domain data needs to be clustered (if they are unlabeled), then they will be treated similar to other speaker classes; in another experiment, we considered all unlabeled data as belonging to only one single class and used it with a 1-VS-1 strategy. On the other hand, unsupervised adapted SVDA does not perform clustering. In every iteration of SVM, unlabeled in-domain data are added to the rest class with no information on their labels. Algorithm 1 summarizes our proposed 1-VS-Rest SVDA. Other advantages of our proposed SVDA includes: SVDA finds the discriminatory directions using the boundary structure of the classes, and also the SVM is a well-known method for small sample size problem (Gu et al., 2010).

Algorithm 1 Algorithm for adapted-SVDA 1-VS-Rest.

C ← Number of speaker classes
X, Y ← i/t/x-vectors, and their labels
N ← Number of all support vectors
γ ← Regularizer parameter $0 \le \gamma \le 1$, and here is set to 0.05.
for $i = 0$ to C **do**
  $X_{cu} = X_i$ concatenate $X_{unlabeled}$
  $Y_{cu} = Y_i$ concatenate $Zeros(0, len(unlabeled))$
  $model = svmtrain(Y_{cu}, X_{cu})$
  Ii = index of SVs for class i
  Ij = index of SVs for unlabeled data
  $w = SVs(Ii) - mean(SVs(Ii))$
  $Vw = Vw + w^T * w$
  $w = SVCoef(Ii) * SVs(Ii) + SVCoef(Ij) * SVs(Ij)$
  $Vb = Vb + w^T * w$
end for
$Vw = (1-\gamma)V_w + \gamma \frac{trace(V_w)}{N-C}$
return eigenvectors corresponding to the k largest eigenvalues of $V_b a = \gamma V_w a$

As stated earlier, we perform the adapted SVDA with a 1-VS-1 strategy, which is summarized in Algorithm 2. We experimented with two different scenarios, (1) all in-domain data are counted as belonging to one speaker class, and (2) we use the pseudo labels estimated from our clustering approach.

Algorithm 2 Algorithm for adapted-SVDA 1-VS-1.

C ← Number of speaker classes
X, Y ← i/t/x-vectors, and their labels
N ← Number of all support vectors
γ ← Regularizer parameter $0 \le \gamma \le 1$, and here is set to 0.05.
$model = svmtrain(Y, X)$
Vw ← Initialize to Zero
for $i = 0$ to C **do**
  Ii = index of SVs for class i
  $w = SVs(Ii) - mean(SVs(Ii))$
  $Vw = Vw + w' * w$
end for
$Vw = (1-\gamma)V_w + \gamma \frac{trace(V_w)}{N-C}$
Vb ← Initialize to Zero
for $i = 0$ to $C-1$ **do**
  for $j = i+1$ to C **do**
    $X_{cu} = X_i$ concatenate X_j
    $Y_{cu} = Y_i$ concatenate Y_j
    $model = svmtrain(Y_{cu}, X_{cu})$
    Ii = index of SVs for class i
    Ij = index of SVs for class j
    $w = SVCoef(Ii) * SVs(Ii) + SVCoef(Ij) * SVs(Ij)$
    $Vb = Vb + w^T * w$
  end for
end for
return eigenvectors corresponding to the k largest eigenvalues of $V_b a = \gamma V_w a$

4.4. PLDA

Here, we perform PLDA adaptation with two different methods, (i.e., supervised and unsupervised adapted PLDA (Garcia-Romero and McCree, 2014; Garcia-Romero et al., 2014)), details are provided in the following description.

For both supervised and unsupervised PLDA adaptation, Γ and Λ parameters, representing the between-class and within-class covariance matrices respectively (Garcia-Romero and McCree, 2014) of PLDA model, need to be updated using the in-domain data. In the supervised adapted PLDA approach, the in-domain data are first clustered (the unlabeled data for SRE-16 & SRE-18 are in-domain data) and when their pseudo labels are estimated, we can perform the traditional PLDA on them. The Λ and Γ parameters of the supervised adapted PLDA are then interpolated as,

$$\begin{aligned} \Gamma_{adapt} &= \alpha \Gamma_{in} + (1-\alpha)\Gamma_{out}, \\ \Lambda_{adapt} &= \alpha \Lambda_{in} + (1-\alpha)\Lambda_{out}. \end{aligned} \tag{10}$$

Here, Γ_{in} and Λ_{in} are the between class and within class covariance matrices for the in-domain data, Γ_{out} and Λ_{out} are the same covariance matrices but calculated from out-domain data. In our experiments, we used $(1-\alpha) = 0.85$.

For unsupervised adapted PLDA, the in-domain data are not clustered first (if they are unlabeled; or their actual labels will not be used if they are labeled). Here, mean and variance of all in-domain data are calculated and used for adapting the PLDA covariance matrices as,

$$\begin{aligned} \Gamma_{adapt} &= \Gamma_{out} + \beta_b S, \\ \Lambda_{adapt} &= \Lambda_{out} + \beta_w S, \end{aligned} \tag{11}$$

where β_b determines the scale for updating the between class covariance toward the excess variance in a particular direction, and β_w is the same but for updating within class covariance matrix. In our experiments, we set $\beta_b = 0.2$ and $\beta_w = 0.6$. In addition, S corresponds to the eigenvalues of adaptation-data (in-domain data) total-covariance in PLDA space (Snyder et al.).

4.5. Score normalization, calibration and fusion

NIST evaluates the performance of each team based on the actCprimary metric. To this end, score calibration is essential. Here, we examine multiple options to prepare scores before fusion, using PAV calibration from the BOSARIS toolkit (Brümmer and De Villiers, 2013) trained with DEV or DEV+Unlabeled to calibrate the scores; or simply linear score fusion of the s-norm scores. Score normalization, specifically s-norm, is used when generating the PLDA score, with an adaptive

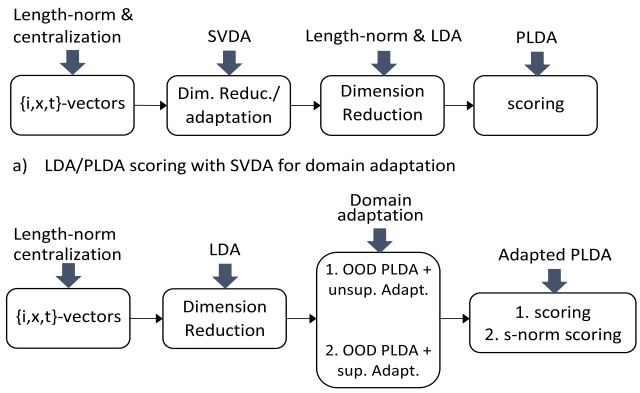

Fig. 2. Flow diagrams of CRSS back-end classifiers (Zhang et al., 2019).

cohort selection scheme followed by a top score selection (Sturim and Reynolds, 2005). In particular, cohorts were selected from the unlabeled DEV set for CMN2 partition in SRE18, and unlabeled DEV set in SRE16.

In order to predict final scores, we combine our multiple single systems. We accomplish this by building a fused model by training a logistic regression model. Let $x = \{x_1, x_2, \ldots, x_n\}$ be a feature vector by concatenating each single system score, where the target variable y is a Bernoulli random variable for which the probability of occurrence is dependent on the prediction given in Eq. (12). Regression coefficients ω are estimated using maximum likelihood estimation. Scores from every single system are combined with the estimated coefficients to obtain the fusion score $\hat{y}$.

$$p(y = 1|x, \omega) = \frac{1}{1 + \exp\left(-\omega^T x\right)} \tag{12}$$

$$\hat{y} = \omega^T x, \tag{13}$$

When the linear weights are learned, the calibrated DEV scores and calibrated EVAL scores are integrated as the final scores to the evaluation.

4.6. Overview of back-end level domain adaptation

Fig. 2 shows the flow diagram of our back-ends with incorporating domain-adaptation methods. Although we carry out experiments on various combinations of domain adaptation techniques (together or separately), this figure summarizes the two different pipelines we find out to be successful and used for our submission to the challenge as well.

Based on our preliminary experiments, we propose performing domain mismatch compensation using either of the pipelines shown in Fig. 2. In other words, domain mismatch is either compensated with SVDA or with adapted PLDA model (which can be supervised adapted PLDA or unsupervised adapted PLDA). In the latter case, scoring can be replaced with s-norm scoring as well. Mean centralization, length normalization and LDA are shared between the two pipelines.

5. Experiments

5.1. Experimental setup

For the UBM/i-Vector framework, we extract 60-dimensional features (20-D MFCC and Δ+$\Delta\Delta$) on a 25 ms window, with a shifting size of 10 ms. Non-speech frames are discarded using an energy-based speech activity detection (SAD). In addition, cepstral mean normalization is applied with a 3-second sliding window. 2048-mixture full covariance UBM and total variability matrix are trained for 600-dimensional i-vector extraction. Next, LDA is used to reduce the dimension of the i-Vectors to 400-D.

In our study, we used the standard Kaldi x-vector recipe to train our baseline x-Vector based system. The input feature vector is a 24-dimensional filter-bank from a 25-ms frame length analysis window, these features are then mean-normalized over a 3-s sliding window. Non-speech segments are removed using an energy-based SAD, though other more advanced SAD methods such as Combo-SAD (Sadjadi and Hansen, 2013) or TO-Combo-SAD (Sadjadi and Hansen, 2013; Ziaei et al., 2014) could also be used for noisy data. The DNN configuration is described in detail in Snyder et al.. The resulting x-vectors are 512 dimensional, which are then reduced to 150-D with LDA.

In the t-vector framework, high-resolution filter bank features are adopted for system development. At the frequency axis, 96-dimensional log mel filter bank features are extracted from a 32-ms analysis speech frame, with a 50% overlap between neighboring frames. Non-speech portions of the utterance are removed using an energy-based SAD. To deal with long-duration samples in the training data, we uniformly segment the speech utterances into 12-second chunks without overlap, which is equivalent to the 750-dimensional feature set along the time axis as the input to the network. To estimate the embedding at the utterance level, we perform segment level embedding averaged in sequential order, in order to obtain the t-Vector. Here, we extract 128-dimensional t-vectors, which are then reduced to 80-D with LDA.

Table 2 summarizes the data used for training each of our developed speaker embedding systems for both SRE-16 and SRE-18 tasks.

Here, SWB includes all Switchboard II phase 2 & 3 and Switchboard Cellular Part 1 & 2 corpora. D2 and D3 listed in Table 2 are augmented by 3-folds after convolving with far-field Room Impulse Responses (RIRs), or by adding noise from the MUSAN corpus (Snyder et al., 2015). The Kaldi x-vector recipe is adopted for this portion of our processing. A speaker filtering criterion is applied to the training dataset as well for t-vector and x-vector feature extraction. For example, 8 min-utt/spk stands for the filtering process that all speakers with less than 8 utterances and less than 500 frames per utterance were excluded for training.

For training the back-end, no augmentation has been applied, our preliminary experiments showed that no gain can be obtained by including augmented data at the back-end training. The out-of-domain PLDA is also trained on only previous SRE data. SVDA, LDA, and PLDA all share the same data. In the experiments where unlabeled data are included in the training of SVDA, LDA, and PLDA, it is explicitly mentioned in the paper. Statistics of the data used for training front-end and back-end stages are summarized in Table 3.

NIST provided scoring software to the participating sites in SRE-16 and SRE-18 to calculate the equal error rate (EER), minimum primary cost (min-Cprimary), and actual primary cost (act-Cprimary). For SRE-16, the software reports both equalized (i.e., false alarm and false reject counts were equalized over various partitions) and unequalized scores.

Table 2
Corpora used in the speaker embedding system training.

Dataset	Copora	Min-Utt/Spk	System
D1	SRE04-08, SWB	1	i-vec
D2	D1+Mixer 6	8	t-vec
D3	D2 + SRE-10 + VoxCeleb	8	x-vec

Table 3
Number of speakers/segments used for training front-end and back-end processing within our speaker recognition system for this study.

System	Front-end	Back-end
i-vector	5756/57273	3794/36410
x-vector	13437/169135	3794/36422
t-vector	5969/132777	3794/36422

Table 4
Ground truth labels (GT) for unlabeled data comparing against pseudo labels estimated with clustering method (CL) in SRE-16 with i-vector/PLDA configuration.

SVDA	LDA	PLDA	DEV		EVAL	
			EER (%)	min-C	EER (%)	min-C
×	GT	GT	17.25/16.76	0.71/0.67	12.64/12.77	0.79/0.8
×	×	×	15.59/16.08	0.70/0.67	12.42/12.68	0.8/0.81
×	CL	CL	16.12/16.34	0.71/0.67	**12.4/12.51**	**0.79/0.79**

Table 5
SVDA domain adaptation with i-vector/PLDA for SRE-16 and SRE-18 tasks.

SVDA	DEV		EVAL	
	EER (%)	min-C	EER (%)	min-C
SRE-16				
No SVDA	15.59/16.08	0.7/0.67	12.33/12.55	0.79/0.8
1-VS-1 (all 1 class)	15.77/16.05	0.7/0.65	10.75/11.04	0.7/0.69
1-VS-1 (CL labels)	15.89/16.32	0.71/0.67	12.33/12.53	0.8/0.8
1-VS-Rest	**15.57/15.95**	**0.66/0.62**	**10.56/10.91**	**0.69/0.68**
SRE-18				
No SVDA	12.17	0.73	12.89	0.78
1-VS-1 (all 1 class)	**10.23**	**0.7**	**11.66**	**0.72**
1-VS-1 (CL labels)	12.07	0.74	12.85	0.77
1-VS-Rest	12.01	0.72	12.92	0.78

In our experiments, we report both equalized/unequalized scores as well. Details on these criteria are provided in Sadjadi et al., NIST (2016, 2018).

5.2. Experimental results

5.2.1. Ground truth labels VS Pseudo labels

Here, we perform experiments to validate how different ways of using unlabeled data can affect the speaker recognition performance. Fig. 3 shows the histogram of both minor and major data for the SRE-16, with ground truth labels as well as the labels estimated with our clustering approach. Table 4 presents the results achieved with ground truth labels vs if we use clustering labels.

The results in Table 4 show that using ground truth labels not only does not improve performance but also degrades the scores (for EVAL specifically). This problem originates from the fact that every speaker has a very small number of samples. Based on the presented histogram for the ground truth and pseudo labels; the latter achieves better performance because there are more samples in each cluster even-though there are errors in labeling. Therefore, we can conclude that having a sufficient number of samples for training LDA and PLDA is important, and with the provided unlabeled data it is better to have either unsupervised adaptation or employ pseudo labels. Throughout the remainder of the experiments, we set aside and never use ground truth labels for the unlabeled data.

5.2.2. SVDA for adaptation

In this subsection, we perform experiments for evaluating the effectiveness of SVDA in domain adaptation. Table 5 summarizes results for i-Vector/PLDA solution for both SRE-16 and SRE-18. Three different SVDA variations have been applied: (1) 1-VS-1 strategy where all unlabeled in-domain data are considered to belong to one cluster; (2) 1-VS-1 where unlabeled data has been clustered first and their clustering (CL) labels used there; and (3) 1-VS-Rest where all unlabeled data are added to the rest class.

Results show that for SRE-16, SVDA achieves +15% and +14% improvement in terms of min-Cprimary and EER respectively. For SRE-18 as well, +8% and +16% improvement were achieved with SVDA in terms of min-Cprimary and EER, respectively. For both SRE-16 and SRE-18, SVDA has been shown to be effective for domain adaptation. In SRE-16, SVDA with 1-VS-Rest strategy and in SRE-18, SVDA with 1-VS-1 (where all unlabeled data are considered to belong to only 1 class) is shown to achieve the better performance comparing against the other SVDA strategies. SVDA with 1-VS-1 and using clustering labels do not provide any improvement. After clustering, the number of samples in each cluster is still small and does not provide an informative structure of the in-domain data, and all samples can be chosen as the support vectors. Therefore, we expect to achieve equivalent performance to LDA, and results also confirm the same. In addition, for SRE-18 because unlabeled data are in the same domain as EVAL set, the mean normalization already is helping; however, for SRE-16 because mean normalization does not provide sufficient adaptation, improvement with SVDA is more clear. Therefore, based on these results, for the

Table 6
Supervised VS Unsupervised PLDA, for SRE-16 and SRE-18.

Adapted PLDA	SRE-18				SRE-16	
	DEV		EVAL		DEV	EVAL
	EER (%)	min-C	EER (%)	min-C	EER (%)	min-C
i-vector						
×	**12.17**	**0.73**	**12.89**	0.78	**12.33/12.55**	**0.79/0.8**
Supervised	15.28	0.78	15.56	0.8	13.93/13.98	0.85/0.84
Unsupervised	14.86	0.73	16.04	**0.76**	13.96/14.23	0.8/0.8
x-vector						
×	11.4	0.78	11.23	0.77	15.32/15.56	0.99/0.99
Supervised	10.34	0.64	11.05	0.65	14.96/15.74	0.97/0.98
Unsupervised	**8.82**	**0.54**	**9.64**	**0.56**	**8.37/8.29**	**0.6/0.61**
t-vector						
×	13.34	0.88	13.87	0.87	17.2/16.15	0.99/0.99
Supervised	11.04	0.76	12.57	0.78	13.16/12.76	0.92/0.93
Unsupervised	**9.5**	**0.53**	**9.62**	**0.67**	**9.17/9.32**	**0.7/0.72**

Table 7
Using data of interest, in-domain data in LDA, SVDA, and PLDA for x-vector, i-vector and t-vector, evaluated on both SRE-16 and SRE-18.

SVDA	LDA	PLDA	SRE-18				SRE-16	
			DEV		EVAL		EVAL	
			EER (%)	min-C	EER (%)	min-C	EER (%)	min-C
i-vector								
×	×	×	12.17	0.73	12.89	0.78	12.42/12.68	0.79/0.81
✓	×	×	12.01	0.71	12.92	0.78	10.66/10.95	0.69/0.69
✓	✓	×	10.76	0.7	12.34	0.76	10.69/11.02	0.69/0.69
✓	×	✓	12.27	0.71	13.05	0.78	12.58/12.77	0.82/0.83
✓	✓	✓	10.91	0.69	12.41	0.76	10.69/10.97	0.7/0.7
×	✓	×	11.15	0.73	12.35	0.75	12.32/12.56	0.77/0.78
×	✓	✓	11.41	0.72	12.47	0.76	12.4/12.51	0.79/0.79
×	×	✓	12.54	0.72	13.06	0.78	12.79/12.95	0.82/0.83
t-vector								
×	×	×	13.34	0.88	13.87	0.87	17.2/16.15	0.99/0.99
✓	×	×	11.93	0.7	10.45	0.74	13.12/12.86	0.89/0.94
✓	✓	×	11.71	0.7	10.4	0.73	12.98/12.91	0.94/0.97
✓	×	✓	9.79	0.57	9.96	0.66	10.01/10.29	0.71/0.72
✓	✓	✓	9.84	0.57	9.99	0.66	10/10.23	0.7/0.72
×	✓	×	13.7	0.89	14.55	0.9	22.43/21.43	0.99/0.99
×	✓	✓	9.52	0.54	9.65	0.67	9.23/9.36	0.71/0.73
×	×	✓	9.49	0.52	9.61	0.67	9.23/9.33	0.7/0.72
x-vector								
×	×	×	11.4	0.78	11.23	0.77	15.32/15.56	0.99/0.99
✓	×	×	8.86	0.61	8.67	0.58	11.45/11.14	0.86/0.89
✓	✓	×	8.89	0.65	8.72	0.59	13.36/12.91	0.99/0.99
✓	×	✓	8.7	0.54	9.96	0.57	8.71/8.46	0.58/0.59
✓	✓	✓	8.66	0.55	9.88	0.56	8.63/8.36	0.58/0.59
×	✓	×	16.97	0.87	13.93	0.86	29.36/27.24	1/1
×	✓	✓	8.55	0.54	9.37	0.56	8.45/8.23	0.62/0.63
×	×	✓	8.8	0.54	9.63	0.56	8.42/8.32	0.6/0.61

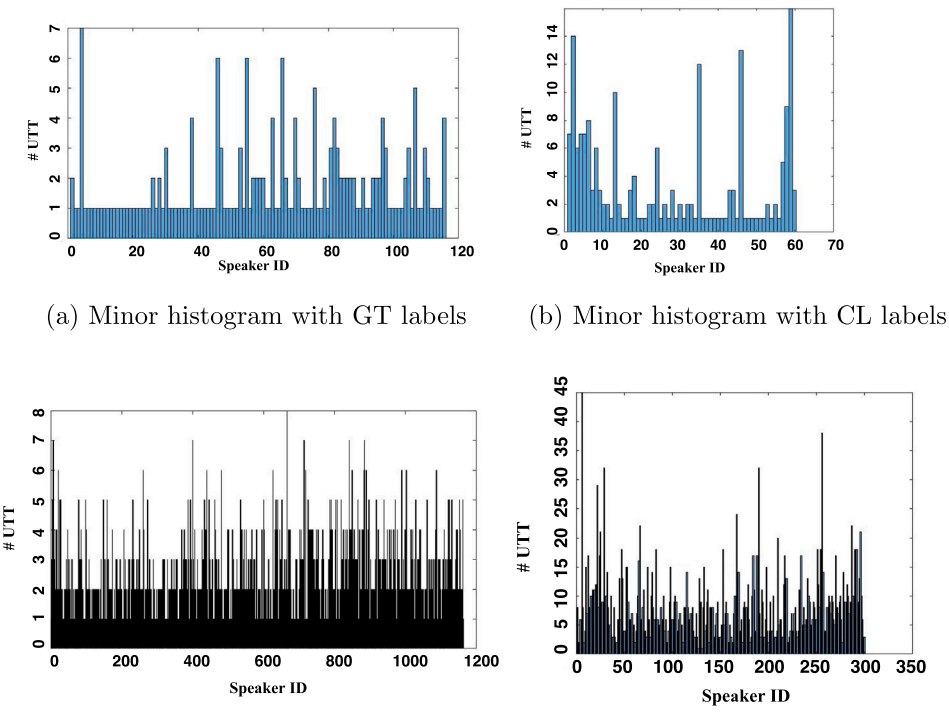

Fig. 3. Histogram for minor and major data (number of utterances per speaker), comparing ground truth labels (GT) and pseudo labels achieved from the clustering approach (CL). For GT labels, there are several speakers with just one utterance (not applicable in adapting LDA/PLDA). The number of speakers with one utterance is reduced with clustering.

Table 8
Score normalization (S-Norm) with supervised PLDA adaptation for x,t,i-vectors evaluated on SRE-18 task.

S-Norm	DEV			EVAL		
	EER (%)	min-C	act-C	EER (%)	min-C	act-C
i-vector						
×	15.57	0.8	**0.91**	16.51	0.81	**0.87**
✓	**13.25**	**0.72**	0.95	**14.19**	**0.77**	1.01
x-vector						
×	10.21	0.62	1.11	10.78	0.64	**0.99**
✓	**8.68**	**0.59**	**0.78**	**9.37**	**0.61**	1.08
t-vector						
×	10.8	0.72	2.2	11.7	0.74	1.67
✓	**9.79**	**0.6**	**0.7**	**10.52**	**0.68**	**0.93**

following experiments, 1-VS-Rest is used for domain adaptation in SRE-16; and 1-VS-1 (all unlabeled data share the same cluster) is applied for SRE-18.

5.2.3. Adapted PLDA

In this section, we compare supervised and unsupervised PLDA adaptation methods for SRE-18 and SRE-16 tasks with i-Vector, t-Vector, and x-Vector embeddings. Results are summarized in Table 6. Here, SVDA is not applied, in order to measure only the effectiveness of adapted PLDA.

For x-Vector and t-Vector embeddings unsupervised adapted PLDA achieves consistent improvement over supervised adapted PLDA and original PLDA. However, for i-Vector, adapted PLDA does not provide any improvement. As it is shown in Snyder et al., augmenting extractor or PLDA does not improve the performance for the i-Vector/PLDA speaker recognition; however, they are very effective for x-Vectors. The discriminative training of x-Vectors and t-Vectors benefit from augmentation techniques is shown here, and further improved with PLDA adaptation. i-Vectors for unlabeled data are not accurate as they have different languages, and i-Vectors trained on only clean (without any augmentation) data, cannot predict the embeddings properly. Therefore, using unlabeled data for adapting the PLDA is shown to degrade performance in the scoring stage. The front-end training and differences between i-Vector, t-Vector, and x-Vector representations are reflected in these scores.

5.2.4. Adaptation results for SRE-16 and SRE-18

In this section, we use in-domain data along with alternate back-end blocks; LDA, SVDA, and PLDA. Results are summarized in Table 7. All systems use in-domain data first for centralization: major data is used for SRE-16 and unlabeled data for SRE-18. For t-Vector and x-Vector, unsupervised PLDA is used where ✓is set for PLDA. LDA needs labeled data for training; therefore, when ✓is on for LDA, the clustered unlabeled data is added to the training set. For SRE-16, 1-VS-Rest SVDA is used and unlabeled data are added to the rest class; for SRE-18 1-VS-1 SVDA (where all unlabeled data are considered to belong to only one class) is used.

The scores for all experiments confirm that domain adaptation at the back-end level is promising, and especially for x-Vectors and t-Vectors, the improvement is more obvious. For i-Vector as well, SVDA is shown to be effective, specifically for SRE-16 where mean centralization is not adequate for domain adaptation. In the i-Vector framework, discriminant analysis and dimension reduction techniques such as SVDA and LDA are shown to be more effective in compensating the domain mismatch rather than the PLDA. However, for x-Vectors and t-Vectors higher gains are achieved by adapting PLDA; however, SVDA still results in better scores. For x-Vector embedding in the SRE-18 task, with SVDA domain adaptation, the EER on the EVAL set is 8.67%, and with an adapted PLDA it is 9.63% which confirms that SVDA is a promising approach to compensate for domain mismatch.

Table 9
Calibration and fusion results where calibration is performed using different data for both SRE-16 and SRE-18. UnLab. refers to in-domain unlabeled data.

Calib.	DEV			EVAL		
	EER (%)	min-C	act-C	EER (%)	min-C	act-C
SRE-16						
DEV	13.81/14.66	0.58/0.55	0.59/0.56	9.41/9.49	0.67/0.66	0.87/0.99
UnLa.+DEV	14.24/14.98	0.59/0.56	0.61/0.58	9.37/9.43	0.65/0.64	0.71/0.81
SRE-18						
DEV	5.63	0.37	0.39	7.14	0.49	0.53
UnLa.+DEV	5.72	0.37	0.38	7.14	0.49	0.53
S-NORM	6.79	0.42	0.43	7.63	0.49	0.5

5.2.5. Score-normalization with S-norm

In this section, the effectiveness of score normalization along with i-vector, t-vector, and x-vector embeddings are studied in terms of EER, min-Cprimary and act-Cprimary for SRE-18. Results are reported in Table 8. The results show that score normalization is effective for all speaker embeddings and in terms of all EER, min-Cprimary and act-Cprimary costs. S-norm incorporates unlabeled data and normalizes the scores which make the fusion of the single systems as well easier; as we can skip the calibration phase when scores are normalized.

5.2.6. Calibration and fusion

Here, we perform experiments to validate the effectiveness of using in-domain data for calibration. When fusing multiple systems, it is necessary to calibrate the scores first or perform a normalization step. Here, in two experiments we calibrated the scores, one with the DEV set, and the other with the unlabeled in-domain set (which they have been clustered first, and then a trial set designed for them). For the second experiment, we did not calibrate the scores, instead, we employ normalized scores for fusion.

For the purpose of highlighting fusion benefits, we use all single systems we developed for our submissions to NIST SRE-16 and SRE-18. Details of these systems are provided in Zhang et al. (2017, 2019). Results for this evaluation are reported in Table 9. For SRE-16 where DEV and EVAL set also have different language sets, the difference between calibrating with only DEV or DEV+Unlabeled data is more obvious. Incorporating in-domain data for calibration helps achieve a closer act-Cprimary cost to min-Cprimary. However, for SRE-18, calibrating with DEV or DEV+in-domain data does not have much of an impact because the DEV set shares the same language sets with EVAL. For SRE-18 as well, the results confirm that score normalization, especially for the EVAL set, performs well. The scores confirm that the fusion of multiple complementary systems significantly outperforms any single system performance.

6. Conclusion

In this study, we have considered multiple domain adaptation methods for speaker recognition with a focus on the NIST SRE-16 and SRE-18 tasks. We developed three alternate speaker embeddings here, i-Vector, t-Vector, and x-Vector. We explored the use of new discriminant analysis with support vectors (SVDA) solution, with new advancements from our previous methods. We evaluated the 1-VS-Rest SVDA strategy for domain adaptation. In addition, a new version of SVDA studied for speaker recognition using unlabeled data; 1-VS-1 where all unlabeled data is considered to belong to one cluster, and 1-VS-1 where unlabeled in-domain data were clustered. Results confirmed that SVDA improves speaker recognition for SRE-16 and SRE-18 by +15% and +8% in terms of min-Cprimary respectively; and in terms of EER +14% and +16% respectively, with i-Vector speaker embeddings. Mean centralization, SVDA, LDA, PLDA, calibration, score normalization, and fusion are phases that we incorporated in-domain data. We developed an effective configuration for each of these steps to properly use the in-domain data. Generally speaking, mean centralization simply provides an effective technique to improve the performance on mismatch data. In addition, unsupervised adaptation is shown to be more effective than the supervised ones (when pseudo labels were incorporated). Assigning labels to in-domain data not only introduces error and negatively affects the performance but also makes speaker clusters smaller and less beneficial when included in LDA/PLDA; pooling all in-domain data together with unsupervised PLDA is shown to perform well for NIST SRE tasks. Using in-domain data with pseudo labels in score calibrations provides a promising solution for domain adaptation.

The results suggest effective steps toward improving domain adaptation for robust speaker recognition. As an insight toward further improvements on domain mismatch compensation for speaker recognition, we suggest incorporating in-domain data at the front-end modeling. At the same time, the methods presented in this paper are applicable to other tasks which suffer from the domain mismatch problem; such as, language recognition or dialect identification tasks where data are recorded under mismatch conditions.

CRediT authorship contribution statement

Fahimeh Bahmaninezhad: Conceptualization, Methodology, Software, Validation, Investigation, Writing - original draft. **Chunlei Zhang:** Methodology, Software, Validation, Investigation. **John H.L. Hansen:** Supervision, Conceptualization, Resources, Writing - review & editing, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the University of Texas at Dallas from the Distinguished University Chair in Telecommunications Engineering held by J.H.L. Hansen, and in part by grants from AFRL FA8750-15-1-0205 (PI: Hansen) and NSF 1918032 (PI: Hansen).

References

Bahmaninezhad, F., Hansen, J.H.L., 2016. Generalized discriminant analysis (GDA) for improved i-vector based speaker recognition. In: ISCA INTERSPEECH, pp. 3643–3647.

Bahmaninezhad, F., Hansen, J.H.L., 2017. i-vector/PLDA speaker recognition using support vectors with discriminant analysis. In: IEEE ICASSP, pp. 5410–5414.

Bahmaninezhad, F., Hansen, J.H.L., 2018. Compensation for domain mismatch in text-independent speaker recognition. In: ISCA INTERSPEECH, pp. 1071–1075.

Bhattacharya, G., Monteiro, J., Alam, J., Kenny, P., 2019. Generative adversarial speaker embedding networks for domain robust end-to-end speaker verification. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 6226–6230.

Borgstrom, B.J., Reynolds, D.A., Singer, E., Sadjadi, O., 2017. Improving the Effectiveness of Speaker Verification Domain Adaptation With Inadequate In-Domain Data. Tech. Rep, MIT Lincoln Laboratory Lexington United States, pp. 1557–1561.

Brümmer, N., De Villiers, E., 2013. The bosaris toolkit: Theory, algorithms and code for surviving the new dcf. arXiv preprint arXiv:1304.2865.

Bulut, A.E., Zhang, Q., Zhang, C., Bahmaninezhad, F., Hansen, J.H.L., 2017. UTD-CRSS Submission for MGB-3 arabic dialect identification: Front-end and back-end advancements on broadcast speech. In: IEEE ASRU Workshop. IEEE, pp. 360–367.

Colibro, D., Vail, C., Dalmasso, E., Farrell, K., Karvitsky, G., Cumani, S., Laface, P., 2017. Nuance-Politecnico di Torino's 2016 NIST speaker recognition evaluation system. In: ISCA INTERSPEECH-2017, pp. 1338–1342.

Dehak, N., Kenny, P.J., Dehak, R., Dumouchel, P., Ouellet, P., 2011. Front-end factor analysis for speaker verification. IEEE Trans. Audio Speech Lang. Process. 19 (4), 788–798.

Fisher, R.A., 1936. The use of multiple measurements in taxonomic problems. Ann. Eugen. 7 (2), 179–188.

Garcia-Romero, D., Espy-Wilson, C.Y., 2011. Analysis of i-vector length normalization in speaker recognition systems. In: ISCA INTERSPEECH, pp. 249–252.

Garcia-Romero, D., McCree, A., 2014. Supervised domain adaptation for i-vector based speaker recognition. In: IEEE ICASSP. IEEE, pp. 4047–4051.

Garcia-Romero, D., McCree, A., Shum, S., Brummer, N., Vaquero, C., 2014. Unsupervised domain adaptation for i-vector speaker recognition. In: ISCA Odyssey: The Speaker and Language Recognition Workshop.

Gu, S., Tan, Y., He, X., 2010. Discriminant analysis via support vectors. Neurocomputing 73 (10), 1669–1675.

Hansen, J.H.L., Hasan, T., 2015. Speaker recognition by machines and humans: A tutorial review. IEEE Signal Process. Mag. 32 (6), 74–99.

Ioffe, S., 2006. Probabilistic linear discriminant analysis. In: European Conference on Computer Vision. Springer, pp. 531–542.

Kenny, P., Boulianne, G., Ouellet, P., Dumouchel, P., 2007. Joint factor analysis versus eigenchannels in speaker recognition. IEEE Trans. Audio Speech Lang. Process. 15 (4), 1435–1447.

Lee, K.A., Hautamäki, V., Kinnunen, T., Larcher, A., Zhang, C., Nautsch, A., Stafylakis, T., Liu, G., Rouvier, M., Rao, W., et al., 2017. The I4U Mega fusion and collaboration for NIST speaker recognition evaluation 2016. In: ISCA INTERSPEECH, pp. 1328–1332.

Lee, K.A., Hautamaki, V., Kinnunen, T., Yamamoto, H., Okabe, K., Vestman, V., Huang, J., Ding, G., Sun, H., Larcher, A., et al., 2019. I4u submission to NIST SRE 2018: Leveraging from a decade of shared experiences. arXiv preprint arXiv:1904.07386.

Li, C., Ma, X., Jiang, B., Li, X., Zhang, X., Liu, X., Cao, Y., Kannan, A., Zhu, Z., 2017. Deep speaker: an end-to-end neural speaker embedding system. arXiv preprint arXiv:1705.02304.

Madikeri, S., Dey, S., Ferras, M., Motlicek, P., Himawan, I., 2016. Idiap Submission to the NIST SRE 2016 Speaker Recognition Evaluation. Tech. Rep., Idiap.

Misra, A., Hansen, J.H.L., 2014. Spoken language mismatch in speaker verification: An investigation with NIST-SRE and CRSS Bi-Ling corpora. In: IEEE SLT: Spoken Language Technology Workshop, pp. 372–377.

Misra, A., Hansen, J.H.L., 2018. Modelling and compensation for language mismatch in speaker verification. Speech Commun. 96, 58–66.

NIST, 2016. NIST 2016 speaker recognition evaluation plan. https://www.nist.gov/sites/default/files/documents/itl/iad/mig/SRE16_Eval_Plan_V1-0.pdf.

NIST, 2018. NIST 2018 speaker recognition evaluation plan.

Plchot, O., Matejka, P., Silnova, A., Novotnỳ, O., Diez, M., Rohdin, J., Glembek, O., Brümmer, N., Swart, A., Jorrın-Prieto, J., et al., 2017. Analysis and description of ABC submission to NIST SRE 2016. In: ISCA INTERSPEECH, pp. 1348–1352.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., et al., 2011. The kaldi speech recognition toolkit. In: IEEE ASRU Workshop. IEEE.

Reynolds, D.A., Quatieri, T.F., Dunn, R.B., 2000. Speaker verification using adapted Gaussian mixture models. Digit. Signal Process. 10 (1–3), 19–41.

Rouvier, M., Bousquet, P.-M., Ajili, M., Kheder, W.B., Matrouf, D., Bonastre, J.-F., 2016. LIA system description for NIST SRE 2016. arXiv preprint arXiv:1612.05168.

Sadjadi, S.O., Hansen, J.H.L., 2013. Unsupervised speech activity detection using voicing measures and perceptual spectral flux. IEEE Signal Process. Lett. 20 (3), 197–200.

Sadjadi, S.O., Kheyrkhah, T., Tong, A., Greenberg, C., Reynolds, E.S., Mason, L., Hernandez-Cordero, J., 2017. The 2016 NIST speaker recognition evaluation. In: ISCA INTERSPEECH, 2017, pp. 1353–1357.

Schroff, F., Kalenichenko, D., Philbin, J., 2015. Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 815–823.

Shon, S., Mun, S., Kim, W., Ko, H., 2017. Autoencoder based domain adaptation for speaker recognition under insufficient channel information. arXiv preprint arXiv:1708.01227.

Shum, S.H., Reynolds, D.A., Garcia-Romero, D., McCree, A., 2014. Unsupervised clustering approaches for domain adaptation in speaker recognition systems.

Snyder, D., Chen, G., Povey, D., 2015. MUSAN: a music, speech, and noise corpus. ArXiv.

Snyder, D., Garcia-Romero, D., Povey, D., Khudanpur, S., 2017. Deep neural network embeddings for text-independent speaker verification. In: ISCA INTERSPEECH, pp. 999–1003.

Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., Khudanpur, S., 2018. X-vectors: Robust DNN embeddings for speaker recognition. In: IEEE ICASSP, 2018.

Sturim, D.E., Reynolds, D.A., 2005. Speaker adaptive cohort selection for Tnorm in text-independent speaker verification. In: IEEE ICASSP, vol. 1, pp. I–741.

Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A., 2017. Inception-v4, inception-resnet and the impact of residual connections on learning. In: AAAI, vol. 4, p. 12.

Torres-Carrasquillo, P.A., Richardson, F., Nercessian, S., Sturim, D., Campbell, W., Gwon, Y., Vattam, S., Dehak, N., Mallidi, H., Nidadavolu, P.S., et al., 2017. The MIT-LL, JHU and LRDE NIST 2016 speaker recognition evaluation system. In: ISCA INTERSPEECH, pp. 1333–1337.

Zhang, C., Bahmaninezhad, F., Ranjan, S., Dubey, H., Xia, W., Hansen, J.H.L., 2019. UTD-CRSS systems for 2018 NIST speaker recognition evaluation. In: IEEE ICASSP, 2019.

Zhang, C., Bahmaninezhad, F., Ranjan, S., Yu, C., Shokouhi, N., Hansen, J.H.L., 2017. UTD-CRSS systems for 2016 NIST speaker recognition evaluation. In: ISCA INTERSPEECH, 2017, pp. 1343–1347.

Zhang, C., Koishida, K., 2017. End-to-end text-independent speaker verification with triplet loss on short utterances. In: ISCA INTERSPEECH.

Zhang, C., Koishida, K., 2017. End-to-end text-independent speaker verification with flexibility in utterance duration. In: IEEE ASRU.

Zhang, C., Koishida, K., Hansen, J.H.L., 2018. Text-independent speaker verification based on triplet convolutional neural network embeddings. IEEE/ACM Trans. Audio Speech Lang. Process. (TASLP) 26 (9), 1633–1644.

Ziaei, A., Kaushik, L., Sangwan, A., Hansen, J.H.L., Oard, D.W., 2014. Speech activity detection for nasa apollo space missions: Challenges and solutions. In: ISCA INTERSPEECH.