DOMAIN ADAPTATION AND INVESTIGATION OF ROBUSTNESS OF DNN-BASED EMBEDDINGS FOR TEXT-INDEPENDENT SPEAKER VERIFICATION USING DILATED RESIDUAL NETWORKS

Seyyed Saeed Sarfjoo Mathew Magimai.-Doss
Sébastien Marcel

Idiap-RR-10-2019

SEPTEMBER 2019

Centre du Parc, Rue Marconi 19, P.O. Box 592, CH - 1920 Martigny
T +41 27 721 77 11 F +41 27 721 77 12 info@idiap.ch www.idiap.ch

Domain Adaptation and Investigation of Robustness of DNN-based Embeddings for Text-Independent Speaker Verification Using Dilated Residual Networks

Seyyed Saeed Sarfjoo
Idiap Research Institute
Rue Marconi 19, 1920 Martigny
saeed.sarfjoo@idiap.ch

Mathew Magimai.-Doss
Idiap Research Institute
Rue Marconi 19, 1920 Martigny
mathew@idiap.ch

Sébastien Marcel
Idiap Research Institute
Rue Marconi 19, 1920 Martigny
marcel@idiap.ch

September 10, 2019

Abstract

Robustness of extracted embeddings in cross-database scenarios is one of the main challenges in text-independent speaker verification (SV) systems. In this paper, we investigate this robustness via performing structural cross-database experiments with or without additive noise. This noise can be added from the *seen* set, where the noise type is similar to the noise which is used in data augmentation for training the SV model, or *unseen* set, where distribution of additive noise in train and evaluation sets are different. For extracting the robust embeddings, we investigate applying the time dilation in the ResNet architecture, so-called dilated residual network (DRN). Dimension and number of segment level layers are tuned in this architecture. The proposed model with time dilation significantly outperformed the ResNet model and is comparable with the state-of-the-art SV systems on Voxceleb1 dataset. In addition, this architecture showed significant robustness in out of domain scenarios.

Language mismatch is part of domain mismatch which recently is one of the main focuses of research in SV systems. Similar to image recognition field, we hypothesize that low-level convolutional neural network (CNN) layers are domain-specific features while high-level CNN layers are domain-independent and have more discriminative power. For adapting these domain-specific units, combination of triplet and intra-class losses are investigated. The adapted model on the evaluation part of the CMN2 dataset, relatively outperformed the DRN and x-vector SV systems without adaptation with 8.0 and 20.5 %, respectively in equal error-rate.

***K*eywords** Speaker verification · Dilated residual network · DRN · Speaker embedding · Domain adaptation.

1 Introduction

For several years, the segment level vector that represents the speech signal, called i-vector, with probabilistic linear discriminant analysis (PLDA) backend, had dominated the text-independent SV research field [1]. In recent years, deep neural networks (DNNs) have shown successful results in several fields including computer vision, speech recognition, or natural language processing [2–4]. Similar to the mentioned fields, DNN-based models successfully have been applied to text-independent SV systems [5–8].

Applying deep learning methods for SV can be done on both frame- or segment-level of the input speech signal. Extracting DNN-based bottleneck features [7], or computing Baum-Welch statistics from DNN for training i-vector [5] represent two frame-level solutions of SV systems. Applying DNN-based non-linear mapping and backend classifier on fixed-length representations (typically i-vectors) is a sample of segment-level solutions for DNN-based SV systems [9]. DNN-based architecture for training the embedding extractor, called semi end-to-end approach, is another segment level solution. In this case, the extracted embeddings can be used for training the back-end classifier (e.g., PLDA) [8, 10], or directly training the speaker-specific classifier [11]. For training the embedding extractor model, recent DNN-based architectures e.g., ResNet with triplet and intra-class loss can be applied [12]. Speaker specific classifier can be trained in a fully end-to-end manner, where "siamese DNNs" are used to approximate the posterior probability of the presented utterances belonging to the same speaker [13].

In many cases, having access to the large high quality in domain dataset is not feasible. Because of this reason, training a robust model is one of the main focuses of research in SV field. Here, we investigate this robustness via performing the structural cross-database experiments with or without additive noise. For additive noise, two conditions are considered (1) where the noise type is similar to the noise which is used in data augmentation for training the SV model, *seen* set, and (2) where distribution of additive noise in train and evaluation sets are different, *unseen* set. Time-delay neural networks (TDNNs) showed significant improvement in the performance of speaker verification systems [8, 14]. This time dilation can be applied to convolutional neural network (CNN) architectures where spectrogram is used as input features. In this paper, for extracting the robust embeddings, we investigate applying the time dilation in the ResNet architecture, so-called dilated residual network (DRN). Number and dimension of segment level layers are tuned in this architecture. As extracting the embeddings with linear activation showed improvement in speaker verification performance [7, 15], bottleneck embedding features (when dimension of the extracted embedding is smaller than the surrounding layers) are extracted after a layer with a linear activation function. For making the verification decision, this extracted embeddings can be used for training the back-end classifier (e.g., PLDA) or directly computing the cosine distance between enrollment and trial embedding vectors. The proposed model significantly outperformed the ResNet model and is comparable with the state-of-the-art SV systems on Voxceleb1 dataset. In addition, this architecture showed significant robustness in out of domain scenarios.

One of the recent challenges in speaker recognition field is domain compensation. In the recent NIST SRE evaluations, language mismatch was one of the main focuses of evaluation. Recently, for alleviating the language mismatch problem, some domain adaptation techniques were proposed [16–19]. In face recognition field, it is shown that high level CNN layers are potentially domain independent and can be used for extracting the embedding and modeling the target identities [20]. On the other hand, low-level CNN layers are domain-specific features and adaptation of these domain-specific units (DSUs) allows to map from the target to the source domain. It is shown that applying the intra-class loss has significant effect on robustness of the triplet loss with respect to noise and intra-class variabilities [12]. In this paper, for adapting the DSUs combination of triplet and intra-class losses is investigated. We investigate the adaptation performance on CMN2 dataset with increasing the number of adaptation layers. As development set of CMN2 dataset is relatively small, adapting the first layer showed the best performance. Based on equal error rate (EER) performance measure, the adapted model relatively outperformed the DRN and x-vector SV systems without adaptation with 8.0 and 20.5 %, respectively. In addition, t-SNE visualization and analysis of heatmap of first CNN layer showed the effectiveness of the proposed method for adaptation. After adaptation, intensity of high frequencies are more similar to the original spectrogram. In addition, represented features from the adapted model shows more similarity for modeling the harmonies and fundamental frequencies.

The rest of this paper is organized as follows: Related works for extracting the DNN-based segment level features and domain adaptation are shown in Section 2. The proposed approaches are discussed in Section 3. Experiment setup and results are discussed in Section 4. Finally, conclusions and future works are shown in Section 5.

2 Related Works

DNN-based architecture for extracting the frame-level embeddings for text-dependent speaker verification was shown in [21]. The average of these frame-level features makes the speaker-specific feature vector, called d-vector. The extracted d-vector from each trial utterance was compared to the d-vector of the enrolled speaker to make a verification decision. The end-to-end system based on d-vector for discriminating between same-speaker and different-speaker pairs was introduced in [22]. However, this end-to-end approach requires a large amount of data to be effective. Feed-forward DNN-based architecture which connects the frame-level and segment-level layers with statistics pooling layer was shown in [23]. This statistics pooling layer computes the concatenation of mean and standard deviation of frame-level layers to be used as an input for segment-level layers. With capturing the long-term speaker characteristics, this architecture can be trained from variable-length speech segments. In [10], siamese architecture was used for training the segment-level embeddings extractor. For improving the robustness of the extracted embeddings, called x-vector [8], the architecture from [23] was trained with data augmentation. Combination of triplet loss and intra-class distance variance regularization for extracting the robust segment-level embeddings was investigated in [12]. In this paper, ResNet architecture with statistical pooling for extracting the segment-level embeddings from spectrogram with two or three seconds chunk size was used. In [24], VGG structure with statistical pooling was used for extracting the segment-level embeddings. In this paper, for training the embedding extractor, combination of softmax and center loss was used. In [15], Shon et al. investigated the discriminability power of frame-level embeddings that are extracted from each hidden layer of CNN-based architecture. They showed that embeddings can be extracted efficiently with linear activation in the embedding layer. Similar to Shon et al., in this paper, we extract embeddings with linear activation in the embedding layer. However, for robustness and effective modeling the sequence of input data, we applied time dilation in the ResNet architecture.

For alleviating the domain shift problem, several domain adaptation approaches have been proposed. In [25], for inter-dataset variability compensation (IDVC) nuisance attribute projection (NAP) was used. In this case, as an i-vector pre-processing step, NAP was used to remove the subspace that represents all variabilities in different datasets. For alleviating the i-vector mismatch across different domains, a domain adversarial technique (DAT) is proposed in [26]. In this paper, for learning a shared feature extractor and two classifiers, multi-task learning framework was used. In this structure, domain-invariant and discriminative features are extracted using a gradient reversal layer in the domain classifier. Different transfer learning strategies when intrinsic neutral/physical mismatch exists between train and evaluation datasets were investigated in [19]. An adversarial method for unsupervised discriminative domain adaptation was proposed in [16]. For reducing the domain mismatch in i-vector and x-vector SV systems, semi-supervised nuisance attribute network (SNAN) was introduced in [17]. In this paper, instead of computing the domain variability from the dataset means, maximum mean discrepancy (MMD) was used as part of the loss function. Our proposed method for domain adaptation is similar to transfer learning method in [19], however in our method, instead of triplet loss, the combination of triplet and intra-class losses is used. In addition, similar to [20] for adapting the DSUs from target to the source domain, low-level CNN layers are investigated.

3 Proposed approaches

The novelty of this paper contains enhanced DNN-based architecture for extracting the embeddings and domain adaptation technique for alleviating the language mismatch. The proposed DRN architecture is shown in Section 3.1. Domain adaptation using triplet and intra-class loss is explained in Section 3.2.

3.1 Dilated Residual Networks

Modeling the sequence of speech frames is a fundamental issue in training the DNN-based models in the majority of speech processing related fields including speaker verification. Using time dilation in CNN, with a fixed size of parameters, discriminative speaker characteristics can be modeled from a larger sequence of speech frames which causes faster convergence time and needs a smaller set for training. DRN architecture was introduced in image classification field for conserving the spacial structure of the scene of the input image in the low dimensional feature map [27]. Although this architecture is capable of modeling the sequence of speech frames, because of keeping the dimension of feature map resolution in inner layers similar to the input layer, in the majority of average size public speech datasets, this architecture will overfit on the training set. Because of this reason we used the original ResNet architecture with dilation in the time domain. The architecture of ResNet which is used in this paper is shown in Figure 1.

Spectrogram from 300 speech frames with 512 FFT resolution was used as input to this architecture. In this condition, the dimension of input for training the DRN model will be 300x257. For applying time dilation, we realized that using a smaller filter size in the first layers will improve the performance of the verification system. Because of this reason, convolution layers with filter size 16, and 32 were used in the first two CNN layers. For modeling the sequence of speech frames, the window size of the first two CNN layers was set to 5 in the time domain. Time dilation was applied on the second and third CNN layers before starting the residual blocks. In these layers, the dilation step was set to 2 and 3, respectively. Similar to [12], the max-pooling layer was applied only on the time domain. Residual blocks follow the same definition in [28]. Using skip connection, fine-grained information of spectrogram can be passed directly to the deeper layers. 3, 4, and 6 residual blocks with 64, 128, and 256 filter size were used, respectively. The dimension of the represented feature in the frequency domain was decreased to 1 using two CNN layers with kernel size 9 in the frequency domain. In these two layers, 256 and 512 were used as filter size, respectively. Mean and standard deviation of represented features was concatenated as statistical pooling layer to create 1024 dimension feature vector as an input to segment-level layers. Similar to [24] for regularization, two dropout layers were used before and after the bottleneck embedding layer. As extracting the embeddings with linear activation showed improvement in speaker verification performance [7, 15], bottleneck embeddings are extracted from the layer with a linear activation function. In the end, using one fully connected layer, the dimension of output features will reach to the number of speaker labels in the training set. Cross-entropy on softmax of output features is used as the loss function. All CNN layers are followed by batch normalization and rectified linear unit (ReLU) activation function. Kernel size of all CNN layers in residual blocks is 3x3 with 1x1 stride and padding. The extracted embeddings are used for training the PLDA backend.

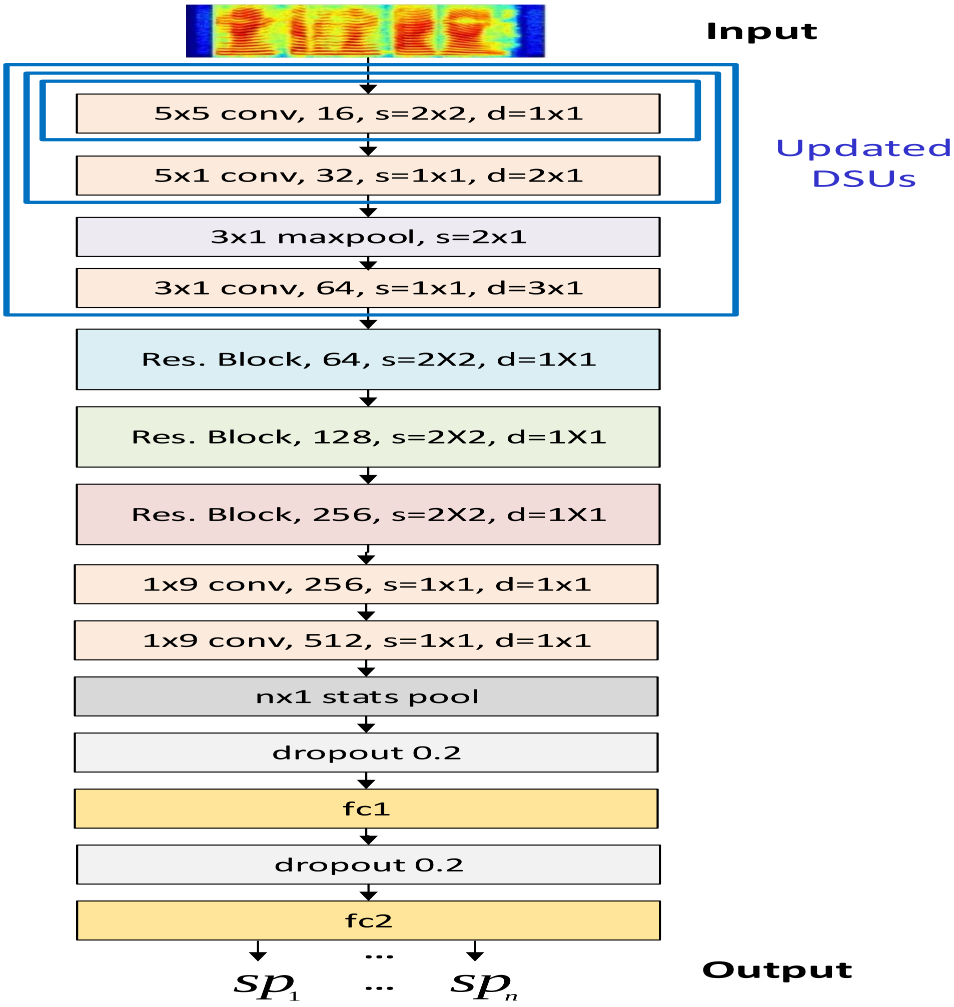

Input
5x5 conv, 16, s=2x2, d=1x1
5x1 conv, 32, s=1x1, d=2x1
3x1 maxpool, s=2x1
3x1 conv, 64, s=1x1, d=3x1
Updated DSUs
Res. Block, 64, s=2X2, d=1X1
Res. Block, 128, s=2X2, d=1X1
Res. Block, 256, s=2X2, d=1X1
1x9 conv, 256, s=1x1, d=1x1
1x9 conv, 512, s=1x1, d=1x1
nx1 stats pool
dropout 0.2
fc1
dropout 0.2
fc2
...
sp_1 ... sp_n
Output

Figure 1: Dilated residual network architecture for training the embeddings extractor. Conv, s, d, fc, and DSUs are convolution layer, stride, dilation, fully connected layer, and domain-specific units, respectively. Embeddings are extracted from the fc1 layer. DSUs are updated from the initial layers.

3.2 Domain adaptation using triplet and intra-class loss

Features that are closer to the input signal (called low level features) assumed to be domain-specific features [20]. On the other hand, features that are closer to the output of DNN architecture are considered to be more task specific and carry more discriminative power. For formalizing this hypothesis, given $X_s = \{x_1, x_2, ..., x_n\}$ and $X_t = \{x_1, x_2, ..., x_m\}$ being a set of samples from source $\mathcal{D}^s$ and target $\mathcal{D}^t$ domains, with theirs set of corresponding labels $Y_s = \{y_1, y_2, ..., y_n\}$ and $Y_t = \{y_1, y_2, ..., y_m\}$, respectively. All parameters of pre-trained CNN-based feature detector from $\mathcal{D}^s$, denoted with θ, can be splitted to domain dependent θ_t and domain independent θ_s parameters, where $P(Y_t|X_s, \theta) = P(Y_t|X_t, [\theta_s, \theta_t])$. These domain dependent parameters θ_t are called domain-specific units. For analyzing the hypothesis that these DSUs are set of low-level features that are correlated to the input signal, we performed the set of experiments on CMN2 dataset. Generic pseudo-code for updating the DSUs in the CNN architecture is shown in Algorithm 1. Position of CNN layers which contains DSUs are shown in Figure 1.

The proposed DRN architecture was used for adaptation. This architecture batch normalizes the signal for every layer. This batch normalization can be defined as:

$$h(x) = \beta_i + \frac{g(W_i \times x) + \mu_i}{\sigma_i}, \tag{1}$$

where β is the batch normalization offset, W is the kernel of CNN layers, g is the non-linear function which is applied to the convolution, usually ReLU, μ and σ are the accumulated mean and standard deviation of the current batch. In the back-propagation step, two variables W and β are updated.

Algorithm 1: Training Strategy Given a Pre-trained CNN-based Model θ, Loss Function $\mathcal{L}$ and the Number of Layers to be Adapted n_{layers}. θ_t is Split Between the CNN Kernel parameter W and the Batch Normalization Offset β

```
Data: $\theta$, $\mathcal{L}$, $n_{layers}$
Result: $\theta_t$
$\theta_t = \theta[: n_{layers}]$; // Domain Spec.  Units
$\theta_s = \theta[n_{layers} :]$; // Domain Indep.  Units
while has_data do
    batch = get_batch() ;
    $\frac{\partial \mathcal{L}}{\partial \theta_t}$ = forward_backward(batch, $\theta$, $\theta_t$, $\mathcal{L}$) ;
    $\theta_t[\beta] = \theta_t[\beta] - \eta \frac{\partial \mathcal{L}}{\partial \theta_t}[\beta]$ ;
    $\theta_t[W] = \theta_t[W] - \eta \frac{\partial \mathcal{L}}{\partial \theta_t}[W]$ ;
end
```

For updating the DSUs combination of triplet and intra-class losses was used. Formally, triplet loss can be defined as:

$$\mathcal{L}_t = \frac{1}{|T|}\sum\Big[d(f(x_a), f(x_p)) - d(f(x_a), f(x_n)) + \alpha\Big]+ \tag{2}$$

where T is the set of all possible triplets of the training set, d is the Euclidean distance in the embedding space, x_a, x_p, and x_n are anchor, positive, and negative samples in training set, respectively. $f(.)$ is the mapping function for extracting the embeddings and α is the margin for computing the triplet loss. For robustness of the triplet loss to noise and intra-class variability, intra-class loss was added to the triplet loss.

Intra-class loss function can be defined as:

$$\mathcal{L}_c(c) = \sum_{x_i, x_j / y_i = y_j = c} \frac{[d(f(x_i), f(x_j)) - \gamma]+}{n_c^2}, \tag{3}$$

where x_i and x_j are the samples from the same class c, n_c is the total number of samples in the current class, and γ is intra-class loss margin. The final loss function can be defined as:

$$\mathcal{L} = \mathcal{L}_t + \frac{\lambda}{K}\sum_c \mathcal{L}_c(c), \tag{4}$$

where λ is intra-class loss weight and K is total number of classes in the current batch.

4 Experimental Setup and Results

The experiments consist of in-domain and domain mismatched experiments. In the in-domain experiments, the train, development, and evaluation sets are selected from the same dataset. For investigating the robustness of the trained model, we performed a series of domain mismatched experiments. Domain mismatch experiments include experiments with and without language mismatch. For investigating the more challenging conditions in experiments without language mismatch, various types of noise with different signal to noise ratios (SNRs) from different datasets can be added to the development or evaluation sets of the target datasets. Finally, the proposed adaptation method was investigated on a more changing dataset with language mismatch. In many cases, having access to the large training datasets is not feasible, because of this reason we focused on experiments with moderately small training sets[1].

4.1 In domain experiments

The in-domain experiments consist of training the proposed DRN model for extracting the bottleneck embeddings and training a PLDA classifier using these embeddings. We performed some experiments for tuning the dimension and number of segment level layers. In the end, we investigate the effect of applying time dilation and data augmentation on the ResNet architecture.

[1] All experiments are reproducible and public repository will be distributed upon acceptance of the manuscript

4.1.1 Training the proposed DRN model

The test set in our experiments is the test set of Voxceleb1 dataset. Voxceleb1 consists of 153,516 utterances from 1,251 speakers where 40 speakers were used as a test set. Train set of Voxceleb1 was used for training the DRN model. One session for each of 1,211 speakers, totally 7,494 utterances, was used as a validation set. On 16-bit single-channel audio streams, spectrograms were generated in a sliding window fashion using a Hamming window of width 25ms and step 10ms. Dropout rate on two dropout layers before and after the embedding layer was set to 20%. The model was trained for 60 epochs and learning rate was set to 1e-3 with decreasing factor of 10 on epoch 30. For all in domain and domain mismatch experiments, RMSprop with weight decay of 1e-4 was used as gradient descent optimizer. Pytorch was used for training and model was trained on GPU GeForce GTX 1080 Ti with 11 GB memory. The extracted embeddings dimension was reduced to 200 after LDA (if the embedding dimension is more than 200), followed by mean subtraction and length normalization before being scored using PLDA classifier. This PLDA training system was implemented using Kaldi toolkit. Equal error rate (EER) and minimum decision cost function (minDCF) with $p_{target} = \{0.01, 0.001\}$ were used as performance measure.

4.1.2 Experiments for tuning the dimension of segment level layer

Investigation of the performance of the proposed system with different embedding dimension using cosine and PLDA scoring on the Voxceleb1 test set is shown in Table 1. Based on EER, embeddings with 60 and 128 dimensions show the best performance with PLDA scoring system. However, based on minDCF, embeddings with 128 dimensions show the best performance. Because of this reason, we chose 128 as an optimal embedding dimension.

Table 1: Performance of the proposed system with different embedding dimension on the Voxceleb1 test set.

Scoring	Dim	EER (%)	$minDCF_{p=0.01}$	$minDCF_{p=0.001}$
Cosine	60	6.1	0.58	0.69
PLDA	60	**5.8**	0.61	0.74
Cosine	128	5.9	**0.56**	**0.66**
PLDA	128	5.8	0.58	0.73
Cosine	200	6.4	0.57	0.72
PLDA	200	6.4	0.58	0.76
Cosine	300	6.5	0.57	0.73
PLDA	300	6.0	0.57	0.70
Cosine	512	6.6	0.58	0.73
PLDA	512	6.3	0.58	0.76

4.1.3 Experiments for tuning the number of segment level layers

Some of recent speaker verification architectures used two segment-level layers when the embeddings can be extracted from the first [8] or second [15] layer. We investigate increasing the number of segment-level layers when the last segment-level layer was used for extracting the embeddings. Result of this investigation is shown in Table 2. Based on these results, increasing the number of segment-level layers will not improve the performance of the speaker verification system.

Table 2: Investigation of increasing the number of segment-level layers on performance of speaker verification system on the Voxceleb1 test set. Dim1 and Dim2 are the dimension of first and second segment-level layers, respectively. (-) in Dim1 is the system with one segment-level layer.

Scoring	Dim1	Dim2	EER (%)	$minDCF_{p=0.01}$	$minDCF_{p=0.001}$
Cosine	128	128	7.1	0.64	0.74
PLDA	128	128	6.5	0.65	0.79
Cosine	256	128	7.5	0.66	0.85
PLDA	256	128	6.7	0.65	0.88
Cosine	512	128	7.1	0.66	0.81
PLDA	512	128	6.8	0.64	0.81
Cosine	-	128	5.9	**0.56**	**0.66**
PLDA	-	128	**5.8**	0.58	0.73

Comparison of the structure of some recent speaker embedding approaches is summarized in Table 3. Similar to [10] in our embedding extractor architecture, spectrogram with a fixed length of three seconds was used as input for training. This chunk of spectrogram was normalized using cepstral mean-variance normalization (CMVN). However, the number of parameters for training the model is reduced to 23M and time dilation was used in 2d-CNN. In contrast to other architectures, in the proposed architecture, except output softmax layer, one fully connected segment-level layer was used. The dimension of the embedding layer was set to 128. Using this small dimension as the bottleneck layer, with decreasing the number of training parameters, in addition to faster convergence, this architecture will prevent over-fitting on the training set. Similar to [15], linear activation function was used before the embedding layer. Global mean subtraction, LDA, length normalization, and PLDA were used as backend processing.

Table 3: Comparing the structure of some recent speaker embedding approaches.

	x-vector [8]	VGG [10]	fl-emb [15]	Ours
Input for training	MFCC	Spectrogram with fixed length (3sec)	MFCC with fixed length (2sec)	Spectrogram with fixed length (3sec)
Input normalization	CMN	CMVN	CMN	CMVN
Structure	TDNN	2d-CNN (VGG-M)	1d-CNN	2d-CNN with time dilation
Parameters	4.4M	64M	13M	23M
Global pooling	Statistics	Average	Statistics	Statistics
Embedding layer	First fully connected layer	Last fully connected layer	Last fully connected layer	Last fully connected layer
Non-linearity	All layers	All layers	All layers except before embedding layer	All layers except before embedding layer
Embedding dimension	512	1024	600	128
Backend Processing	Zero-mean norm.+LDA +length norm.+PLDA	Euclidean Distance with Siamese network	Zero-mean norm.+LDA +length norm.+PLDA	Zero-mean norm.+LDA +length norm.+PLDA

4.1.4 Investigation of applying time dilation and data augmentation

To increase the amount and diversity of the existing training data, Voxceleb1 dataset was augmented with additive noise and reverberation. For reverberation and noise, similar to [8] RIR, and MUSAN datasets were used, respectively. RIR is the collection of room impulse responses measured in the different room sizes. The MUSAN dataset consists of over 900 noise samples, 42 hours of music from various genres and 60 hours of speech from twelve languages. These datasets are freely available[2]. For data augmentation, the clean version of speech samples mixed with some noise, randomly chosen from four different categories. These noise categories contain *babble, music, noise,* and *reverb* which are speech, music, noise, and room impulse response, respectively. In the first three categories, the selected noises from MUSAN dataset are added to the original speech in different SNR levels. In the last category, the training recording has artificially reverberated via convolution with simulated RIRs. In average, for each clean sample, two noisy samples were generated and the proposed system is trained using this augmented dataset.

For investigation of the effect of adding time dilation to the ResNet architecture, we performed some experiments with the proposed ResNet architecture without time dilation. These experiments are done with and without data augmentation. The comparison of the performance of the proposed systems with some recent speaker verification systems on the test set of Voxceleb1 dataset is shown in Table 4. Including the time dilation to the ResNet architecture, relatively outperformed the ResNet model with 4.9% when data augmentation is not included to the train set. With data augmentation, DRN model outperformed the ResNet model with 4.0%, relatively. Based on EER, the proposed DRN method outperformed the recent methods in speaker verification. The difference between $minDCF$ of the proposed method and recent systems is negligible. According to the observed results, including the time dilation to the ResNet architecture will improve the performance of the speaker verification system and the performance of this architecture can be comparable with current DNN-based speaker verification methods.

4.2 Domain mismatch experiments without language mismatch

For investigation of the robustness of the proposed method, the performance of the pre-trained model was investigated in cross-database scenarios. The list of selected datasets and the number of speakers and utterances for each of training, development, and evaluation subsets are shown in Table 5. For this experiment, Voxforge[3] and Mobio[4] datasets were selected. VoxForge was set up to collect transcribed speech to use with free and open source speech recognition engines and Mobio database consists of bi-modal (audio and video) data taken from 152 people to use with biometric systems. The speech in Voxforge dataset is cleaner than Mobio dataset. Two subsets of Voxforge dataset, when 10 or 100 speakers

[2] http://www.openslr.org

[3] http://www.voxforge.org/home/downloads

[4] https://www.idiap.ch/dataset/mobio

Table 4: Comparison of the performance of speaker verification systems on the Voxceleb1 test set. Systems trained with data augmentation are labeled with *. RN and DRN are the proposed ResNet and Dilated ResNet models, respectively. All baseline systems are reported from the mentioned references except an i-vector system as a reproducible baseline.

System	EER (%)	$minDCF_{p=0.01}$	$minDCF_{p=0.001}$
i-vector	5.4	$\mathbf{0.45}$	0.63
i-vector* [15]	5.5	0.48	$\mathbf{0.61}$
VGG [10]	7.8	0.71	-
Pretrained+Intra. [12]	7.9	0.72	0.95
x-vector [15]	7.1	0.57	0.75
x-vector* [15]	6.0	0.53	0.75
fl-emb [15]	5.9	0.50	0.62
fl-emb* [15]	5.3	$\mathbf{0.45}$	0.63
VGG+center. [24]	4.9	-	-
RN	6.1	0.6	0.67
RN*	5.0	0.53	0.71
DRN	5.8	0.58	0.73
DRN*	$\mathbf{4.8}$	0.51	0.78

Table 5: Number of speakers and utterances for each selected datasets: training, development, and evaluation.

Dataset	Train (#spkrs/#utts)	Dev (#spkrs/#utts)		Eval (#spkrs/#utts)	
		Enroll	Trial	Enroll	Trial
Voxforge 10	10/3148	10/1304	3000	10/1509	3000
Voxforge 100	100/12886	100/1000	231700	100/1000	245900
Mobio male	37/7881	24/240	60480	38/380	151620
Mobio female	13/2769	18/180	34020	20/200	42000

were selected in each of train, development, and evaluation sets, were used in this experiment. The Mobio dataset was split based on the gender of speakers. For investigation of the robustness of the proposed method, in addition to clean sets, some experiments with additive noise from seen and unseen datasets were done.

4.2.1 Cross-database experiment using clean sets

In this experiment, the performance of the pre-trained model was compared with some speaker verification methods when these models were trained using the train set of the selected datasets. In this experiment, GMM-UBM, Ivec-Cosine, and Ivec-PLDA were used as some baseline speaker verification systems. The goal of this experiment is to investigate the performance of the proposed method on cross-database scenarios when databases with various type and size were used. As selected datasets are moderately small sets, cosine similarity was used as a scoring measure between extracted embeddings from enrollment and probe trials. In the case of multi-segment enrollment, the average of extracted embeddings was used as the enrollment feature vector. For this cross-database experiment, Bob signal-processing and machine learning toolbox was used[5]. The result of this investigation on the clean cross-database scenario is shown in Table 6.

Based on minDCF performance measure, the proposed DRN method outperformed the baseline systems on all selected datasets. Based on EER performance measure on Voxforge 10 dataset, DRN model significantly outperformed the baseline systems. However, with increasing the size of the training set to 100 speakers, GMM-UBM method showed better performance. Even in this condition, the result of the proposed method is comparable with the baseline systems. In Mobio male dataset, the proposed system significantly outperformed the baseline systems on both development and evaluation sets. Based on EER performance measure, on the development set of Mobio female dataset, DRN method significantly outperformed the baseline systems. Although the result of DRN method is comparable with the baseline systems on the evaluation set of this dataset, GMM-UBM method showed better performance. Based on these results, the performance of the proposed DRN model is comparable with the baseline systems in cross-database scenarios.

[5] https://www.idiap.ch/software/bob

Table 6: Investigation of the robustness of the proposed system in clean cross-database scenario without language mismatch.

Dataset	System	Dev Set			Eval Set		
		EER (%)	$minDCF_{p=0.01}$	$minDCF_{p=0.001}$	EER (%)	$minDCF_{p=0.01}$	$minDCF_{p=0.001}$
Voxforge 10	GMM-UBM	1.9	0.08	0.08	1.9	0.32	0.66
Voxforge 10	Ivec-Cosine	5.3	0.57	0.64	14.5	0.79	0.79
Voxforge 10	Ivec-PLDA	10.0	0.98	0.99	13.6	0.93	0.93
Voxforge 10	DRN	**0.6**	**0.02**	**0.02**	**1.5**	**0.09**	**0.09**
Voxforge 100	GMM-UBM	**2.3**	0.37	0.74	**2.8**	0.43	0.85
Voxforge 100	Ivec-Cosine	3.1	0.30	0.60	**2.8**	0.40	0.75
Voxforge 100	Ivec-PLDA	6.1	0.73	0.97	5.8	0.76	0.98
Voxforge 100	DRN	2.6	**0.21**	**0.49**	3.0	**0.23**	**0.39**
Mobio male	GMM-UBM	18.2	0.96	0.98	11.5	0.91	0.98
Mobio male	Ivec-Cosine	13.6	0.88	0.97	10.7	0.80	0.95
Mobio male	Ivec-PLDA	18.8	0.98	0.98	13.2	0.96	0.99
Mobio male	DRN	**7.3**	**0.68**	**0.81**	**8.3**	**0.76**	**0.92**
Mobio female	GMM-UBM	21.4	0.93	0.98	**15.6**	0.98	0.99
Mobio female	Ivec-Cosine	19.3	0.98	0.99	19.0	0.98	0.99
Mobio female	Ivec-PLDA	23.3	1.00	1.00	25.7	1.00	1.00
Mobio female	DRN	**7.7**	**0.76**	**0.89**	16.4	**0.94**	**0.97**

Table 7: Investigation of the robustness of the proposed system in cross-database scenario with additive noise from seen sets without language mismatch.

Dataset	System	Dev Set (with/without noise on enroll)			Eval Set (with/without noise on enroll)		
		EER (%)	$minDCF_{p=0.01}$	$minDCF_{p=0.001}$	EER (%)	$minDCF_{p=0.01}$	$minDCF_{p=0.001}$
Voxforge 10	GMM-UBM	9.0/8.7	0.62/0.59	0.68/0.60	9.6/14.6	0.66/0.55	0.66/0.79
Voxforge 10	Ivec-Cosine	19.3/17.3	0.91/0.81	0.97/0.81	23.3/25.3	0.95/0.91	0.95/0.91
Voxforge 10	Ivec-PLDA	25.3/24.0	0.99/0.97	0.99/0.97	24.6/26.0	0.98/0.98	0.98/0.98
Voxforge 10	DRN	**1.0/1.2**	**0.15/0.16**	**0.30/0.37**	**2.6/3.2**	**0.25/0.26**	**0.35/0.33**
Voxforge 100	GMM-UBM	11.6/10.9	0.88/0.78	1.0/0.96	12.6/11.1	0.92/0.79	1.0/0.98
Voxforge 100	Ivec-Cosine	11.6/11.1	0.82/0.77	0.95/0.95	11.7/12.7	0.83/0.84	0.97/0.98
Voxforge 100	Ivec-PLDA	18.1/18.1	0.99/0.99	0.99/0.99	19.1/19.6	0.99/0.99	0.99/0.99
Voxforge 100	DRN	**3.8/3.7**	**0.39/0.35**	**0.66/0.63**	**4.5/4.6**	**0.38/0.36**	**0.62/0.58**
Mobio male	GMM-UBM	27.9/25.1	1.0/0.99	1.0/0.99	26.0/20.6	1.0/0.98	1.0/0.99
Mobio male	Ivec-Cosine	24.0/22.6	0.99/0.96	0.99/0.99	20.7/20.4	0.97/0.95	0.99/0.99
Mobio male	Ivec-PLDA	27.7/25.5	1.0/0.99	1.0/0.99	24.5/22.4	0.99/0.99	0.99/0.99
Mobio male	DRN	**10.4/9.9**	**0.84/0.92**	**0.96/0.82**	**11.6/11.1**	**0.80/0.82**	**0.94/0.97**
Mobio female	GMM-UBM	28.1/25.8	0.99/0.98	0.99/0.99	29.0/23.4	0.99/0.99	0.99/0.99
Mobio female	Ivec-Cosine	25.9/24.9	1.0/0.99	1.0/0.99	26.9/25.8	0.99/0.99	0.99/0.99
Mobio female	Ivec-PLDA	35.3/31.2	1.0/1.0	1.0/1.0	33.4/31.9	1.0/1.0	1.0/1.0
Mobio female	DRN	**12.6/10.0**	**0.85/0.82**	**0.96/0.94**	**22.3/19.4**	**0.97/0.96**	**0.97/0.97**

4.2.2 Cross-database experiment using additive noise from seen sets

For investigation of the robustness of the proposed method on more challenging sets, development and evaluation set of the selected datasets were augmented with random noise from MUSAN set. For each clean sample in these sets, randomly one, two, or three samples from one of background *music*, or *noise* subsets with a random signal to noise ratio (SNR) level (5-15 dB SNR) was added to the clean signal. These selected samples from MUSAN set are different from the samples which are used for creating the data augmentation set for training the proposed model. Because of adding noise from the same set, we performed the separate experiments for investigation of the robustness of the proposed method using additive noise from the seen sets. The result of this investigation is shown in Table 7.

For this investigation, two experiments were done: using clean or noisy enrollment data. In this experiment, the performance of the proposed method for all datasets significantly outperformed the baseline systems. On Voxforge 10 dataset, adding noise significantly deteriorate the performance of the baseline systems. However, for DRN system, this additive noise slightly deteriorates the performance of the system. The same pattern was observable for Voxforge 100. However, with increasing the training size of the datasets, baseline systems show more robustness to the additive noise.

4.2.3 Cross-database experiment using additive noise from unseen sets

For investigation of robustness of the proposed method using additive noise from unseen sets, development and evaluation set of the selected datasets were augmented with random noise from DEMAND set.[6] DEMAND is a collection of multi-channel recordings of acoustic noise in various environments. For each clean sample in development and evaluation set of selected datasets, randomly one or two samples from one of background *metro*, *car*, *bus*, *river*, *washing*, or *park* subsets with random SNR level (0-15 dB SNR) was added to the clean signal. The result of this investigation is shown in Table 8.

Table 8: Investigation of the robustness of the proposed system in cross-database scenario with additive noise from unseen sets without language mismatch.

Dataset	System	Dev Set (with/without noise on enroll)			Eval Set (with/without noise on enroll)		
		EER (%)	$minDCF_{p=0.01}$	$minDCF_{p=0.001}$	EER (%)	$minDCF_{p=0.01}$	$minDCF_{p=0.001}$
Voxforge 10	GMM-UBM	9.6/13.0	0.72/0.67	0.79/0.68	14.6/17.0	0.84/0.72	0.96/0.72
Voxforge 10	Ivec-Cosine	18.0/17.6	0.93/0.87	0.93/0.87	22.9/25.6	0.99/0.98	0.99/0.98
Voxforge 10	Ivec-PLDA	24.2/25.3	0.98/1.0	0.98/1.0	23.6/29.6	0.98/1.0	0.98/1.0
Voxforge 10	DRN	**2.0/3.2**	**0.19/0.39**	**0.47/0.40**	**4.0/5.3**	**0.23/0.45**	**0.23/0.58**
Voxforge 100	GMM-UBM	13.0/15.1	0.86/0.87	0.95/0.98	13.3/15.8	0.91/0.87	0.97/0.98
Voxforge 100	Ivec-Cosine	11.7/13.5	0.81/0.88	0.96/0.98	12.5/15.2	0.80/0.87	0.96/0.98
Voxforge 100	Ivec-PLDA	19.1/23.9	0.99/0.99	1.0/0.99	19.1/24.6	0.99/0.99	0.99/0.99
Voxforge 100	DRN	**4.4/6.1**	**0.44/0.49**	**0.7/0.78**	**5.3/7.2**	**0.44/0.50**	**0.67/0.69**
Mobio male	GMM-UBM	28.4/29.0	0.99/0.98	0.99/0.99	26.3/21.2	0.99/0.96	0.99/0.99
Mobio male	Ivec-Cosine	22.1/21.4	0.98/0.97	0.99/0.98	18.6/18.9	0.97/0.91	0.99/0.99
Mobio male	Ivec-PLDA	26.8/24.3	0.99/0.99	0.99/0.99	22.9/20.9	0.99/0.98	0.99/0.99
Mobio male	DRN	**10.5/11.5**	**0.93/0.91**	**0.97/0.97**	**11.7/13.2**	**0.87/0.90**	**0.96/0.97**
Mobio female	GMM-UBM	28.1/29.4	0.99/0.96	0.99/0.98	30.7/27.3	0.99/0.99	0.99/0.99
Mobio female	Ivec-Cosine	26.8/23.8	0.99/0.99	0.99/0.99	27.4/26.9	0.99/0.99	0.99/0.99
Mobio female	Ivec-PLDA	33.3/28.6	1.0/1.0	1.0/1.0	32.2/31.1	1.0/0.99	1.0/0.99
Mobio female	DRN	**13.4/13.6**	**0.90/0.93**	**0.93/0.96**	**22.0/23.0**	**0.97/0.97**	**0.97/0.97**

Similar to Section 4.2.2, experiments with clean and noisy enrollment data were done and similarly, the performance of the proposed method for all datasets significantly outperformed the baseline systems. With respect to additive noise from the seen sets, additive noise from the unseen set will slightly deteriorate the performance of the DRN system. In this condition, using noisy enrollment data will improve the performance of the proposed method.

4.3 Domain mismatch experiments with language mismatch

For investigation of the robustness of the proposed system in cross-database scenario with language mismatch, the performance of the DRN system was investigated on CMN2 dataset. CMN2 is part of NIST SRE 2018 which contains conversational telephone speech in Tunisian Arabic language. For handling the sampling rate mismatch between train and evaluation sets, models are trained with both 8 and 16 kHz. In the 16 kHz experiments, telephone speech data are up-sampled to 16 kHz. For mapping the target domain to the source domain, DSUs in the initial layers are updated using combination of triplet and intra-class losses on the extracted embeddings from the DRN model. For adaptation, labeled part of development set of CMN2 dataset was used. In this experiment, α, γ, and λ were set to 0.2, 0.2, and 1e-3, respectively. The batch size was set to 60 with 10 unique speaker labels per batch. The model was trained for 90 epochs and learning rate was set to 1e-3 with decreasing factor of 10 on every 30 epochs. In this experiment, the effect of increasing the number of adaptation layers (from one layer to three layers) is investigated. For a fair comparison with the baseline system, x-vector was trained using augmented train part of Voxceleb1 dataset. Investigation of the robustness of the proposed systems in cross-database scenario with language mismatch on the evaluation set of CMN2 dataset is shown in Table 9

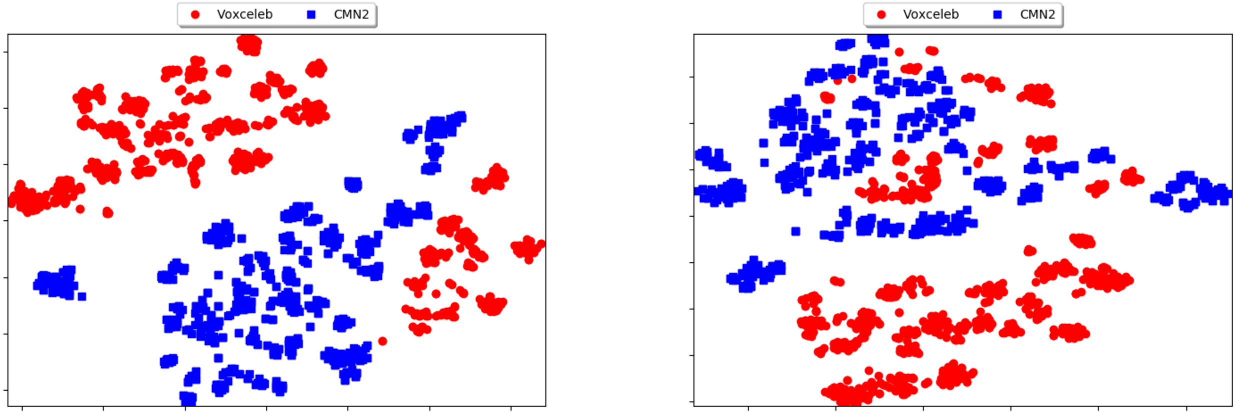
Because of using the relatively small English set for training the models, with respect to other experiments, language mismatch experiment is more challenging. For eliminating the sampling frequency mismatch, 8 kHz models outperformed the 16 kHz ones, however, these models will not generalize well in the language mismatch scenario. Without adaptation, the proposed DRN model outperformed the x-vector model in both 8 and 16 kHz experiments. Updating the DSUs improved the performance of DRN system, however increasing the number of adaptation layers did not further improve the performance. Shortage of labeled data in development set of CMN2 dataset is one of the reasons for this observation.

[6] https://zenodo.org/record/1227121

Table 9: Investigation of the robustness of the proposed systems in cross-database scenario with language mismatch on the evaluation set of CMN2 dataset. Systems are trained using the augmented train part of the Voxceleb1 dataset. -16k and -8k are experiments that performed on 16 and 8 kHz samples, respectively. -1layer to -3layers are adaptation systems with updating the DSUs from one to three layers, respectively.

System	EER (%)	$minDCF_{p=0.01}$	$minDCF_{p=0.001}$
x-vector-16k	34.4	0.98	0.99
DRN-16k	30.1	0.99	0.99
DRN-16k-1layer	**28.1**	**0.96**	**0.97**
DRN-16k-2layers	29.6	0.98	0.98
DRN-16k-3layers	29.6	0.98	0.99
x-vector-8k	30.1	0.98	0.99
DRN-8k	26.0	0.97	0.98
DRN-8k-1layer	**23.9**	**0.96**	**0.97**
DRN-8k-2layers	25.8	0.96	0.98
DRN-8k-3layers	26.6	0.97	0.99

For analyzing the effect of updating the DSUs on domain adaptation, we visualized the extracted embeddings from evaluation set of 8 kHz version of CMN2 and Voxceleb1 datasets. This visualization is done before and after adaptation of the first CNN layer. For dimensionality reduction, t-distributed stochastic neighbor embedding (t-SNE) technique was applied. The 2D visualization is shown in Figure 2. For this experiment, 1000 samples from evaluation sets of Voxceleb1 and CMN2 datasets were randomly selected. For CMN2 dataset, samples are selected from target trials. Based on the observed patterns, for both source and target datasets, intra-speaker distance for extracted embeddings is smaller than inter-speaker distance. The embeddings from each target speaker, create a cluster in the low dimensional t-SNE space. In addition, we can observe that after adaptation, extracted embeddings from Voxceleb1 and CMN2 datasets are closer to each other.

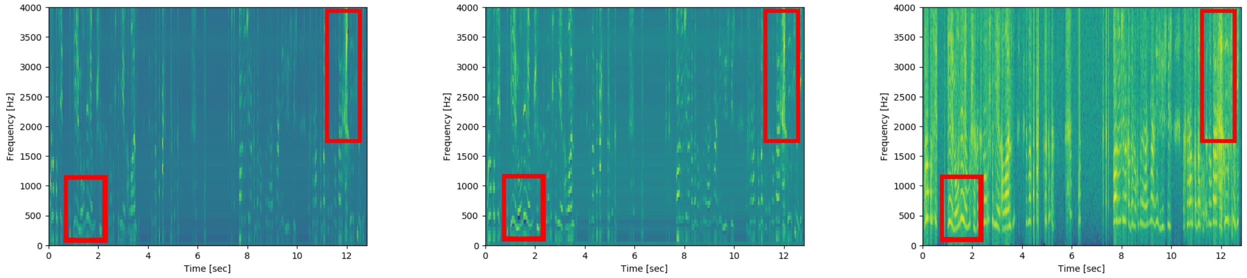


(a) Before adaptation

(b) After adaptation

Figure 2: t-SNE visualization of extracted embeddings from Voxceleb1 and CMN2 datasets, before and after adaptation.

For further analysis, we plot the heatmap [29] of represented features from the first CNN layer, before and after adaptation. Comparison of the heatmap of represented feature for a random sample “ihwwadns_sre18” from CMN2 dataset is shown in Figure 3. Because of using stride in the first CNN layer, dimension of the represented feature in both temporal and spectral domains is half of the original spectrogram. After linear normalizing the extracted features, similar to spectrogram, we plot the natural logarithm of the normalized features. From the plotted heatmaps, we can observe that the trained DNN model for discriminating the speakers focuses on harmonies and frequencies with high intensity. After adaptation, the intensity of high frequencies is more similar to the original spectrogram. In addition, represented features from the adapted model show more similarity for modeling the harmonies.



(a) Before adaptation

(b) After adaptation

(c) Original spectrogram

Figure 3: Heatmap of first CNN layer before and after adaptation for a sample from CMN2 dataset. The region for comparing the intensity of high frequencies and harmonies are shown with bold rectangles.

5 Conclusions and Future Works

In this paper, we investigated applying the time dilation in the ResNet architecture for SV systems. Dimension and number of segment level layers were investigated in this architecture. For investigating the robustness of the proposed method, organized cross-database experiments were performed with additive noise from seen or unseen sets. The proposed model on Voxceleb1 dataset relatively outperformed the DRN and x-vector systems with 4.0 and 20.0 %, respectively in EER. In addition, this architecture showed significant robustness in out of domain scenarios.

One of the main focuses of research in SV field is domain adaptation for reducing the language mismatch. Similar to image recognition field, we hypothesized that low-level CNN layers are domain-specific features while high-level layers are domain-independent and have more discriminative power. For adapting these domain-specific units, we investigated transfer learning method with combination of triplet and intra-class losses on extracted embeddings from DRN architecture. The adapted model on evaluation part of CMN2 dataset, relatively outperformed the DRN and x-vector SV systems without adaptation with 8.0 and 20.5 %, respectively in EER. In addition, t-SNE visualization and analysis of heatmap of the first CNN layer showed the effectiveness of the proposed method for adaptation. Based on initial experiments, updating the DSUs from initial layers was more effective than updating them from the final layers on the current DNN architecture. In the future, we will investigate the effect of adaptation on each layer of DNN architecture. In addition, investigation of combination of supervised and unsupervised methods for cross-lingual adaptation in SV systems can be direction of research in this field.

6 Acknowledgements

This work was partially supported by (1) the European H2020 project TeSLA and by (2) the SNF project Unified Speech Processing Framework for Trustworthy Speaker Recognition (UniTS).

References

[1] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet. Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):788–798, 2011.

[2] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, pages 6645–6649. IEEE, 2013.

[3] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[4] Alexis Conneau, Holger Schwenk, Loıc Barrault, and Yann Lecun. Very deep convolutional networks for natural language processing. *arXiv preprint*, 2016.

[5] Yun Lei, Nicolas Scheffer, Luciana Ferrer, and Mitchell McLaren. A novel scheme for speaker recognition using a phonetically-aware deep neural network. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 1695–1699. IEEE, 2014.

[6] Patrick Kenny, Vishwa Gupta, Themos Stafylakis, Pierre Ouellet, and Jahangir Alam. Deep neural networks for extracting baum-welch statistics for speaker recognition. In *Proc. Odyssey*, pages 293–298, 2014.

[7] Alicia Lozano-Diez, Anna Silnova, Pavel Matejka, Ondrej Glembek, Oldrich Plchot, Jan Pešán, Lukáš Burget, and Joaquin Gonzalez-Rodriguez. Analysis and optimization of bottleneck features for speaker recognition. In *Proceedings of Odyssey*, volume 2016, pages 352–357, 2016.

[8] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur. X-vectors: Robust dnn embeddings for speaker recognition. *Submitted to ICASSP*, 2018.

[9] Timur Pekhovsky, Sergey Novoselov, Aleksei Sholohov, and Oleg Kudashev. On autoencoders in the i-vector space for speaker recognition. In *Proc. Odyssey*, pages 217–224, 2016.

[10] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. Voxceleb: a large-scale speaker identification dataset. *arXiv preprint arXiv:1706.08612*, 2017.

[11] Hannah Muckenhirn, Mathew Magimai Doss, and Sébastien Marcell. Towards directly modeling raw speech signal for speaker verification using CNNs. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4884–4888. IEEE, 2018.

[12] Nam Le and Jean-Marc Odobez. Robust and discriminative speaker embedding via intra-class distance variance regularization. In *Proceedings Interspeech*, pages 2257–2261, 2018.

[13] David Snyder, Pegah Ghahremani, Daniel Povey, Daniel Garcia-Romero, Yishay Carmiel, and Sanjeev Khudanpur. Deep neural network-based speaker embeddings for end-to-end speaker verification. In *Spoken Language Technology Workshop (SLT), 2016 IEEE*, pages 165–170. IEEE, 2016.

[14] David Snyder, Daniel Garcia-Romero, and Daniel Povey. Time delay deep neural network-based universal background models for speaker recognition. In *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, pages 92–97. IEEE, 2015.

[15] Suwon Shon, Hao Tang, and James Glass. Frame-level speaker embeddings for text-independent speaker recognition and analysis of end-to-end model. *arXiv preprint arXiv:1809.04437*, 2018.

[16] Wei Xia, Jing Huang, and John HL Hansen. Cross-lingual text-independent speaker verification using unsupervised adversarial discriminative domain adaptation. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5816–5820. IEEE, 2019.

[17] Weiwei Lin, Man-Wai Mak, Youzhi Tu, and Jen-Tzung Chien. Semi-supervised nuisance-attribute networks for domain adaptation. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6236–6240. IEEE, 2019.

[18] Gautam Bhattacharya, Joao Monteiro, Jahangir Alam, and Patrick Kenny. 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)*, pages 6226–6230. IEEE, 2019.

[19] Chunlei Zhang, Shivesh Ranjan, and John Hansen. An analysis of transfer learning for domain mismatched text-independent speaker verification. In *Proc. Odyssey 2018 The Speaker and Language Recognition Workshop*, pages 181–186, 2018.

[20] Tiago de Freitas Pereira, André Anjos, and Sébastien Marcel. Heterogeneous face recognition using domain specific units. *IEEE Transactions on Information Forensics and Security*, 2018.

[21] Ehsan Variani, Xin Lei, Erik McDermott, Ignacio Lopez-Moreno, and Javier Gonzalez-Dominguez. Deep neural networks for small footprint text-dependent speaker verification. In *ICASSP*, volume 14, pages 4052–4056. Citeseer, 2014.

[22] Georg Heigold, Ignacio Moreno, Samy Bengio, and Noam Shazeer. End-to-end text-dependent speaker verification. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 5115–5119. IEEE, 2016.

[23] David Snyder, Daniel Garcia-Romero, Daniel Povey, and Sanjeev Khudanpur. Deep neural network embeddings for text-independent speaker verification. In *Proc. Interspeech*, pages 999–1003, 2017.

[24] Sarthak Yadav and Atul Rai. Learning discriminative features for speaker identification and verification. *Proc. Interspeech 2018*, pages 2237–2241, 2018.

[25] Hagai Aronowitz. Inter dataset variability compensation for speaker recognition. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4002–4006. IEEE, 2014.

[26] Qing Wang, Wei Rao, Sining Sun, Leib Xie, Eng Siong Chng, and Haizhou Li. Unsupervised domain adaptation via domain adversarial training for speaker recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4889–4893. IEEE, 2018.

[27] Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. Dilated residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 472–480, 2017.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[29] Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, 28(11):2660–2673, 2016.