# Template-matching for Text-dependent Speaker Verification

Subhadeep Dey[a], Petr Motlicek[a], Srikanth Madikeri[a], Marc Ferras[a]

[a] *Idiap Research Institute, Martigny, Switzerland*

**Abstract**

In the last decade, i-vector and Joint Factor Analysis (JFA) approaches to speaker modeling have become ubiquitous in the area of automatic speaker recognition. Both of these techniques involve the computation of posterior probabilities, using either Gaussian Mixture Models (GMM) or Deep Neural Networks (DNN), as a prior step to estimating i-vectors or speaker factors. GMMs focus on implicitly modeling phonetic information of acoustic features while DNNs focus on explicitly modeling phonetic/linguistic units. For text-dependent speaker verification, DNN-based systems have considerably outperformed GMM for fixed-phrase tasks. However, both approaches ignore phone sequence information. In this paper, we aim at exploiting this information by using Dynamic Time Warping (DTW) with speaker-informative features. These features are obtained from i-vector models extracted over short speech segments, also called online i-vectors. Probabilistic Linear Discriminant Analysis (PLDA) is further used to project online i-vectors onto a speaker-discriminative subspace. The proposed DTW approach obtained at

*Email addresses:* subhadeep.dey@idiap.ch (Subhadeep Dey), motlicek@idiap.ch (Petr Motlicek), srikanth.madikeri@idiap.ch (Srikanth Madikeri), marc.ferras@idiap.ch (Marc Ferras)

least 74% relative improvement in equal error rate on the RSR corpus over other state-of-the-art approaches, including i-vector and JFA.
*Keywords:* Text-dependent speaker verification, DNN posteriors, Dynamic Time Warping

---

## 1. Introduction

Text-independent Speaker Verification (SV) is concerned with the verification of a claimed identity against a speech recording without constraints. For several years, the i-vector framework (Dehak et al., 2011) has been considered as state-of-the-art approach in text-independent SV. Together with discriminative techniques such as Linear Discriminant Analysis (LDA) and Probabilistic Linear Discriminant Analysis (PLDA), i-vectors are able to compactly and efficiently represent speaker identity from speech recordings. However, the performance obtained by i-vector systems is still not satisfactory in many conditions, especially for short recordings (Motlicek et al., 2015). Another considerable drawback is the need for enormous amounts of data for training models. Nevertheless, by constraining the speaker to utter a specific content, also called text-dependent mode of authentication, the performance can be expected to increase considerably.

Phrase-based text-dependent SV involves the authentication of a claimed identity against a speaker speaking a known phrase. This phrase can be speaker-specific or common to all speakers and the phrase spoken by the speaker during enrollment phase may be different from the test phrase. In this work, we consider the scenario where the phrases chosen by the system during testing have already been uttered by the speaker during enrollment.

Accepting a claim involves recognizing both the speaker (based on its acoustic characteristics) and the phrase content of a speech utterance. In other words, impostor trials can be divided into three categories: (i) the content (phrase) does not match, (ii) the speaker does not match, and (iii) neither the speaker nor content matches.

State-of-the-art text-dependent SV systems are able to exploit text constraints to obtain high recognition accuracy (Kenny et al., 2014a; Dey et al., 2016a). These systems are inspired by text-independent techniques such as i-vector and Joint Factor Analysis (JFA) (Kenny et al., 2014b; Novoselov et al., 2014; Zeinali et al., 2015) being tailored to the text-dependent SV task. Besides intra-speaker and inter-session variabilities, text-dependent SV systems also need to deal with content variability. Short utterance durations pose common problems as well.

Content or linguistic information is relevant to text-dependent SV systems as accept/reject decisions are directly linked to it. Content information has been introduced into conventional SV systems in different ways. Phoneme-dependent Gaussian Mixture Model - Universal Background Models (GMM-UBM) were used to extract speaker-adapted mean supervectors that were later classified using Support Vector Machines (SVM) (Zhang et al., 2007). Recent work uses Deep Neural Network (DNN) based Sufficient Statistics (SS) to compute i-vectors (Lei et al., 2014). Unlike conventional GMM-UBM, DNNs are trained in a supervised manner using phonetic classes obtained after forced alignment of the training data usually with Hidden Markov Model (HMM)/GMM acoustic models of Automatic Speech Recognition (ASR) system.

Motivated by the above results, our recent work explored various modeling frameworks such as i-vectors and JFA by applying SS extracted using a DNN for text-dependent SV (Dey et al., 2016a). Experimental carried out on the RSR database (Larcher et al., 2014) indicate superior performance of the JFA approach (Dey et al., 2016a). Even though JFA explicitly models phonetic content for text-dependent task, sequence information for the content variability is still ignored. Considering that content information can be decomposed into phonetic units (PU) and its sequence, i.e. the phone sequence information (PSI), standard i-vector and JFA systems obtain the same verification score for any permutation of the PSI. For the phrase "OK Google", which comprises the sequence of phones /əʊˈkeɪˈguːgᵊl/, the permutation /ˈguːgᵊləʊˈkeI/ would be expected to obtain the same score. This is due to the fact that SS depend only on the average feature characteristics in the i-vector and JFA frameworks. In this paper, we aim at exploiting both PU and PSI.

In Su and Wegmann (2016), sequence information is partially used in an i-vector system by computing SS from the HMM/DNN based ASR decoder (employing a Language Model (LM) in addition to acoustic modeling). The posteriors obtained after decoding, which is designed to model long-term temporal information of the speech signal, are more sparse than the posteriors directly estimated by DNN acoustic model.

An alternative to exploit sequence information explicitly using template matching technique, i.e. Dynamic Time Warping (DTW), which has shown to perform well for text-dependent SV (Jelil et al., 2015; Dey et al., 2016a). Compared to applying conventional spectral features in the DTW algorithm,

posteriors extracted from DNN and GMM-UBM have been successfully used. It has been observed that DTW using DNN posterior features provides good performance in the "content-mismatch" conditions (Dey et al., 2016a). However, this system performed poorly in the "speaker-mismatch" condition, probably due to content-discriminative features being computed using a DNN. In this condition, the i-vector and JFA systems performed better (Dey et al., 2016a).

In this paper, we extend our earlier work on DTW-based systems (Dey et al., 2016a) and propose to incorporate speaker-informative features generated by an i-vector system. Although conventional i-vector systems are usually applied over long utterances (2.5 mins) in SV tasks, it has been shown that computing i-vectors from short segments of speech (also termed as online i-vectors) can also contain sufficient speaker information for the speaker diarization task (Madikeri et al., 2015). In this work, we propose to use online i-vectors for text-dependent SV by estimating sequences of online i-vectors computed over the whole utterance. The DTW algorithm is used as a backend, matching enrollment and test online i-vector sequences. Since both PU and PSI are incorporated in this approach, better speaker recognition performance is expected compared to baseline systems exploiting PU only. PLDA model is further trained to discriminate the speaker-content variability of the online i-vectors. The model was used to obtain speaker-content projected i-vectors to be used in the DTW algorithm. These techniques are evaluated on the fixed-phrase parts of the RedDots (Lee et al., 2015) and RSR (Larcher et al., 2014) corpora, both designed for text-dependent SV.

In this paper, two different approaches to text-dependent SV (model-

based and sequence-based) are described. The paper is organized as follows: model-based SV approaches, (i-vector, JFA) are described in Section 2. DTW approaches are described in Section 3. Section 4 introduces the experimental setup for evaluating the SV systems and Section 5 presents the results for the proposed systems. Finally, conclusions are presented in Section 6.

## 2. Speaker Modeling approaches

The conventional approaches to text-dependent SV system for characterizing speakers is based on GMM based techniques. It assumes that the speaker data is generated from a GMM. In this work, we describe three techniques to model speakers, namely, (i) Maximum-a-Posteriori (MAP), (ii) i-vector, and (iii) JFA, which are referred to as model-based SV systems. The MAP models speaker by a set of Gaussians, which are obtained by adapting the UBM. The subspace approaches (i-vector and JFA) assume that the invariant speaker characteristics to lie in a low dimensional subspace. The speakers are represented by a fixed-dimension vector in the subspace.

### *2.1. GMM based baseline system*

The model-based SV techniques (MAP, i-vector and JFA) computes SS from a GMM. The zeroth order ($N_k$) and first order ($\mathbf{f}_k$) statistics of an utterance $\mathbf{O} = \{ \mathbf{o}_1, \mathbf{o}_2, \cdots, \mathbf{o}_{\mathrm{M}}\}$ are given by the following equations

$$N_k = \sum_t \gamma_{k,t} \tag{1}$$

$$\mathbf{f}_k = \sum_t \gamma_{k,t} \mathbf{o}_t, \tag{2}$$

where $\gamma_{k,t}$ is the posterior probability of $k^{th}$ Gaussian unit given feature $\mathbf{o}_t$. These SS are then used by the MAP, i-vector and JFA approaches for estimating the parameters of the respective models.

### *2.1.1. MAP*

In the MAP framework, a GMM, also referred to as the UBM, is estimated by pooling data from all the speakers (Reynolds et al., 2000). To enroll a speaker, the training data is used to adapt the parameters of the UBM with respect to the MAP criterion. In practice, adapting only the means has been shown to be sufficient. The mean of the adapted-GMM is a linear interpolation of UBM mean supervector and first order statistics ($\mathbf{f}_k$ from Equation 2). To verify a claim against a speaker the likelihood of the utterance is computed with respect to the adapted-GMM. This technique can be effective in conditions where there is limited or no speaker labels to estimate parameters of PLDA (in the i-vector system) or JFA models.

### *2.1.2. I-vector system*

In the i-vector framework, the mean supervector of an utterance is transformed using a low dimension total variability matrix, as given by the following equation

$$\mathbf{s} = \boldsymbol{\mu} + \mathbf{T}\mathbf{w}, \tag{3}$$

where $\mathbf{s}$ is the mean supervector, $\boldsymbol{\mu}$ is the mean supervector of the UBM. The matrix $\mathbf{T}$ defines a low rank projection of the mean supervectors. The low-dimensional projections, $\mathbf{w}$, are called i-vectors.

Estimating the i-vector representation of an utterance involves computing the zeroth and first order statistics with respect to the UBM (Glembek

et al., 2011) (Equations 1 and 2). The statistics accumulate information over the entire speech recording thereby losing much of the content information that can be important for text-dependent speaker verification. The content variability is modelled by explicitly training a back-end classifier, usually a PLDA model, with multiple examples of speaker-phrase combinations (Larcher et al., 2014; Dey et al., 2016a). In a simplified PLDA model (Romero and Wilson, 2011), an i-vector ($\mathbf{w}$) can be decomposed into speaker factors ($\mathbf{h}$) and channel effects as follows:

$$\mathbf{w} = \mathbf{m} + \mathbf{V}\mathbf{h} + \boldsymbol{\epsilon}, \tag{4}$$

where $\mathbf{m}$ is the mean of the i-vectors, $\mathbf{V}$ is the speaker subspace and $\boldsymbol{\epsilon}$ is the residue term that captures inter-session variabilities.

Data used to train the PLDA plays a critical role in determining the performance of the speaker verification system. Modeling PLDA requires multiple speaker-phrase combinations from many speakers. Moreover for the models to be generalizable, data consisting of a large variety of speaker-phrases are required. Otherwise, the model can overfit to a specific set of speakers or phrases.

*2.1.3. JFA system*

JFA can be used as an alternative to the i-vector-PLDA approach mentioned earlier for text-dependent speaker verification by explicitly modeling the content variability as a separate factor (Kenny et al., 2014c,a). The JFA model is given as follows

$$\mathbf{s} = \boldsymbol{\mu} + \mathbf{D}\mathbf{z} + \mathbf{U}\mathbf{x} \ , \tag{5}$$

where $\mathbf{D}$ is a diagonal matrix capturing the speaker variabilities, $\mathbf{z}$ is the corresponding latent vector representing the speaker-phrase, $\mathbf{U}$ is the Eigenchannel matrix and $\mathbf{x}$ is the corresponding channel factor representing the channel effects of a speech recording. The hyperparameters, $\mathbf{D}$ and $\mathbf{U}$, are estimated based on the Expectation Maximization (EM) algorithm (Kenny et al., 2005). Given the hyperparameters, we use the Gauss-Seidel approach (Vogt et al., 2005; Vogt and Sridharan, 2008) to obtain estimates of $\mathbf{z}$ and $\mathbf{x}$ for a speech recording.

### *2.2. DNN based system*

The parameters and latent factors in the i-vector and JFA models are estimated using the posteriors from a GMM. In the past, several studies have suggested that integrating linguistic information into speaker recognition systems can be useful (Lei et al., 2014; Motlicek et al., 2015; Park and Hazen, 2002; Sturim et al., 2002; Baker et al., 2005). In HMM/DNN automatic speech recognition (Lei et al., 2014), state posterior probabilities are obtained at the output of the DNN acoustic model. These are used to compute SS using the feature vectors of an utterance. This approach achieved significant improvements over a baseline i-vector system (Lei et al., 2014). This suggests that i-vectors benefit from the acoustic space being partitioned by well-defined linguistic units. Clearly, this is difficult to achieve using unsupervised training, as used for GMM-UBM estimation.

After the successful integration of DNNs in the i-vector text-independent system, we explored its application to text-dependent systems. Indeed, the same approach could be readily applied to JFA systems as well. The use of DNN in the MAP approach has not been studied in the literature. It is

beyond the scope of this paper to explore techniques for incorporating DNN in the MAP framework for SV.

### *2.2.1. HMM/DNN ASR system*

In ASR, the acoustic models are context-dependent tied states (Povey et al., 2011) (also called senones), obtained using a decision tree based on contextual and data-driven criteria. A HMM/GMM system provides the state alignment for the training data, used to extract state labels for DNN training. The DNN uses a final softmax output layer aims at estimating the posterior probabilities of such tied states from input features. Given the large number of DNN outputs, the estimated posterior vectors tend to be sparse. A major drawback of training such a DNN is the need for a large amount of transcribed data. On the other side, posteriors for well defined linguistic units are obtained.

Although HMM/DNN system provides state-of-the-art ASR performance in matched condition, there is still a significant gap in performance for mismatched conditions (Huang et al., 2014). In literature, to address the domain mismatch problem (Gemello et al., 2006; Li and Sim, 2010), the acoustic model is adapted to the evaluation condition using a small amount of transcribed data. In a DNN framework, it is usually done by adapting the weights of one of the layer keeping others layers fixed. The weights of the last layer of the DNN are adapted using a limited amount of transcribed domain data with the senone-discriminative backpropagation algorithm. The adapted-DNN provides better ASR results on the evaluation data than the DNN trained in resource rich domain. Thus we believe that the better ASR system will help in SV process.

## 3. Template matching

DNN-based approaches to i-vector/JFA modeling use PU information as target classes. However, the PSI of the phrase is ignored. We believe that exploiting the PSI in addition to PU will further improve performance, as text constraints for the task are being considered. One approach to implicitly use PSI in i-vector system is by estimating senone posteriors obtained from after ASR decoding. These posteriors capture the long term context of speech signal as it is computed from decoded output (using LM and lexical model) (Su and Wegmann, 2016).

An alternative method to use the PSI is to model the idiosyncrasies of the speaker. A speaker not only has distinctive acoustic features but uses language in a characteristic manner, also called idiosyncrasies (Amino et al., 2006)). These distinctive patterns of the speaker are usually expressed in terms of usage of words, phonemes (Shriberg, 2007; Campbell et al., 2003). In Campbell et al. (2003), PSI was used to estimate phone N-gram frequency. However, these approaches are mainly used as a source of high-level speaker-dependent features. As such, they have been used to enhance the performance of acoustic-based SV systems.

In a different direction, the spectral vectors of the speech signal, consisting of a specific phone sequence, have been used with DTW algorithm (Dey et al., 2016a; Jelil et al., 2015). This approach was shown to be effective for matching sequence of features and outperforms the model-based SV systems in content-mismatch conditions (Dey et al., 2016a), while in speaker mismatch condition, it provides reasonable accuracy. Motivated by the achieved results and the fact that DTW has not been investigated well enough after

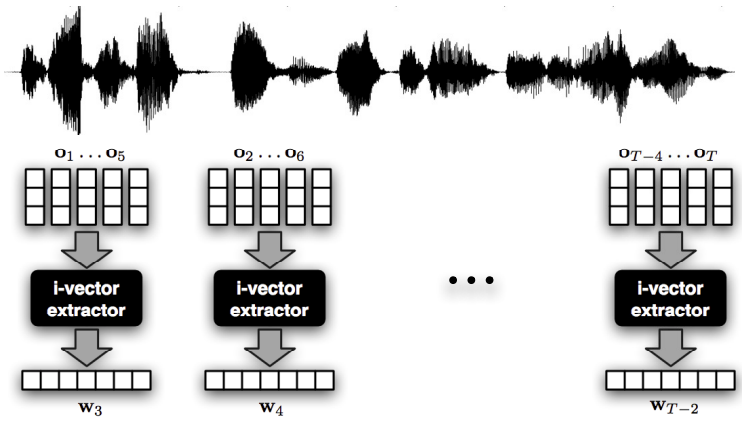


Figure 1: Extraction of online i-vectors.

the emergence of subspace based techniques, we intend to further explore the DTW technique to address the text-dependent SV problem.

### *3.1. DTW system*

The DTW algorithm is a dynamic programming technique to compute the distance between two sequences of spectral vectors of arbitary length, and is commonly applied in query-by-example spoken term detection and other data mining tasks (Rodriguez-Fuentes et al., 2014; Keogh and Ratanamahatana, 2005). Being a non-parametric approach, it is well-suited for limited- or zero-resource tasks (Versteegh et al., 2015). The algorithm takes two sequences of features as input and finds the minimum cost mapping between them. The procedure involves computing all possible local distance between the two sequences (within a given range) and then back-tracking along the optimal path in terms of minimum distance (Brown and Rabiner, 1982). The DTW system performs well for the text-dependent SV task, especially for content-mismatch trials, due to the constraint in the spoken phrase (Dey et al., 2016a).

In a conventional DTW system, MFCCs are used as input features to the

DTW algorithm for performing text-dependent SV (Ramasubramanian et al., 2006). Besides MFCCs, senone posteriors have also been used as features to the algorithm (Dey et al., 2016a) by replacing Euclidean distance by the Kullback-Leibler (KL) divergence measure. Impressive gains were obtained with respect to a state-of-the-art i-vector system on content-mismatch conditions, while on speaker-mismatch trials, the system performs reasonably well (Dey et al., 2016a). As expected, the results indicate that these features might not contain enough speaker information to address a speaker recognition task. In the speaker-mismatch condition, the i-vector and JFA approaches performed considerably better than the DTW system. In view of these results, we propose to introduce speaker-informative features in the DTW algorithm. An i-vector system is used to extract these features. As opposed to the conventional approach of estimating i-vector for a whole utterance (2.5 mins for text-independent and 3 s for text-dependent systems), we propose to compute i-vectors on short segments of speech around 200ms. These features have also been referred to as online i-vectors (Peddinti et al., 2015; Madikeri et al., 2015).

#### *3.1.1. Online i-vector features*

The online i-vector features have been recently used for speech recognition and speaker diarization tasks, where it has shown promising results (Peddinti et al., 2015; Madikeri et al., 2015). In ASR, online i-vectors have been used for the purpose of adapting neural networks to speakers (Peddinti et al., 2015). In this case, online i-vectors are used as an input to the neural network, in addition to spectral features, to enhance speaker-specific information. The results obtained by this approach indicate that online i-vectors contain suf-

ficient speaker information to improve ASR performance.

Online i-vectors have also been applied for the speaker diarization task within the Information Bottleneck (IB) framework for speaker clustering (Madikeri et al., 2015; Vijayasenan et al., 2011; Tishby et al., 2000). In this work, online i-vectors were appended to MFCC features to be fed into the speaker clustering algorithm. The additional gain in performance obtained by this approach compared to using only the spectral features suggests that the online i-vector representation carries speaker information as well. Motivated by the progress in content and speaker oriented tasks, we propose using online i-vectors as features for DTW systems. We now proceed to describe the method to apply online i-vectors.

Figure 1 illustrates the process of extracting online i-vectors from the speech signal. Let the speech utterance contains 'M' frames of speech given by $\mathbf{O} = \{\mathbf{o}_1, \cdots, \mathbf{o}_\mathrm{M}\}$, where $\mathbf{o}_t$ is the $t^{th}$ speech frame. The online i-vector corresponding to $t^{th}$ speech frame of an utterance is computed with a context size of L frames. The SS are computed on the sequence of speech frames, starting from $t$ - L to $t$ + L, for obtaining $t^{th}$ feature vector. For a context size L = 10 frames, a sliding window of 21 frames is used with a shift step of 1 frame. Windows are centered at each frame in the utterance, which results in fewer frames being considered at the utterance boundaries. The corresponding sequence of online i-vectors is represented by $\mathbf{W} = \{\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_\mathrm{M}\}$ for an utterance. To compare two sequences of online i-vectors, the DTW algorithm is used with the cosine distance metric as given by the following equation

$$d(\mathbf{w}_i, \mathbf{w}_j) = 1 - \frac{\mathbf{w}_i' \mathbf{w}_j}{||\mathbf{w}_i||\ ||\mathbf{w}_j||},$$

where $\mathbf{w}_i$ and $\mathbf{w}_j$ are two i-vectors, $d(\mathbf{w}_i, \mathbf{w}_j)$ is the cosine distance between them and $||.||$ represents the vector norm.

DTW scores computed on online i-vectors are expected to reflect both content and speaker similarities between enrollment and test templates. A window length of 200ms, corresponding to average syllable duration, is able to capture both types of information.

*3.1.2. PLDA projection features*

A channel compensation model, such as PLDA, is usually applied on top of i-vectors in text-independent SV systems. The PLDA model produces verification scores by comparing two i-vectors. We apply the PLDA model on top of online i-vectors as we believe that it will help to factor out unnecessary channel information from the features. Training a PLDA model for the SV task uses speaker labels to define a set of classes to be discriminated. It is common to have multiple instances of speaker labelled i-vectors available for large text-independent datasets (Romero and McCree, 2014; Lei et al., 2014). For a text-dependent scenario, the outcome of the task is linked to identifying content and speaker. This motivates the use of speaker-content classes for PLDA training (Dey et al., 2016a; Larcher et al., 2014). Besides labeling content as whole phrases, phone classes can be obtained from a forced alignment of the data against given transcripts as well. Speaker labels are typically available as meta-data provided as part of the dataset. In this work, we experiment with both speaker-phrase and speaker-phone labels for training the PLDA hyperparameters on online i-vectors. PLDA is usually

trained with speaker-phrase labels for text-dependent SV task (Dey et al., 2016a; Larcher et al., 2014). We now describe the training procedure for PLDA with speaker-phone labels only.

The sequence of online i-vector features is extracted for $q^{th}$ utterance of speaker $s_k$, which is represented by $\mathbf{W}_q^{s_k} = \{\mathbf{w}_{1,q}^{s_k}, \mathbf{w}_{2,q}^{s_k}, \cdots, \mathbf{w}_{\mathrm{M},q}^{s_k}\}$. The HMM/DNN based ASR system is used to align the speech signal with respect to the senone classes, which are then mapped to obtain the phone labels. We create a set of P phone classes for the speaker ($s_k$) ($\{D_1^{s_k}, D_2^{s_k}, D_3^{s_k}, \cdots, D_P^{s_k}\}$) for training the PLDA model, with the online i-vector $\mathbf{w}_t^{s_k} \in D_r^{s_k}$ if $t^{th}$ MFCC feature of the utterance is aligned to $r^{th}$ monophone. In a database with S speakers, we have S×P classes for training the PLDA model.

DTW uses online i-vectors after projection onto the inter-class PLDA subspace, also called PLDA projections. The cosine distance between enrollment and test templates is used for this purpose. In this process, PLDA compensates for variabilities other than speaker-content, such as channel variability. The PLDA projections have been successfully used in related speech processing tasks such as speaker diarization and domain adaptation (Dey et al., 2016b; Madikeri et al., 2015). A reasonable gain in performance for speaker diarization is observed as compared to the system using only i-vector, which suggests that the PLDA model has enhanced the speaker representation of i-vectors (Madikeri et al., 2015).

The PLDA projection features are obtained as follows. From the PLDA model of Equation 4, the probability distribution of the speaker factor is given by the following equation

$$p(\mathbf{h}|\mathbf{w}) = \mathcal{N}(\mathbf{m}^{'}, \mathbf{C}), \tag{6}$$

where the $\mathbf{m}^{'}$ is the mean and $\mathbf{C}$ is the covariance matrix of the Gaussian distribution. The mean is given by

$$\mathbf{m}^{'} = \mathbf{C}\mathbf{V}\mathbf{\Sigma}^{-1}(\mathbf{w} - \mathbf{m}), \tag{7}$$

where $\mathbf{\Sigma}$ is the covariance matrix of the error term of Equation 4 and $\mathbf{I}$ is the identity matrix. The covariance matrix ($\mathbf{C}$) is given by

$$\mathbf{C} = (\mathbf{I} + \mathbf{V}'\mathbf{\Sigma}^{-1}\mathbf{V})^{-1}.$$

In this work, we refer the mean of the Gaussian distribution ($\mathbf{m}^{'}$) as the PLDA projection feature (the point estimate of the posterior distribution of the speaker factor), which is subsequently applied in the DTW framework. The PLDA projection vector of a frame of speech is obtained by first computing the online i-vector and then projecting in the PLDA subspace as given by the Equation 7. Thus for an utterance, the number of PLDA-projection features is same as the speech frames.

## 4. Experimental Setup

In this section, we describe the experimental setup for the baseline and the proposed systems, the configurations of the i-vector PLDA, JFA and the HMM/DNN based ASR systems. The systems are evaluated on RSR and RedDots database according to the protocol in Larcher et al. (2014); Lee et al. (2015). Following the setup in Larcher et al. (2014); Dey et al. (2016a), the system performance is evaluated on three conditions, labeled as Cond1, Cond2, Cond3 and an additional condition (Cond-all) with the trials from all

three conditions put together. In condition 1, each trial is associated with determining if the phrases are the same or different. In condition 2, the system is required to differentiate speakers pronouncing the same content. In condition 3, both the speaker and the phrase can be different. Performance for the systems are presented in terms of Equal Error Rate (EER) and minimum Decision Cost Function (minDCF) with the probability of target being 0.01, cost of false alarm error probability being 1 and cost of miss error probability being 10.

### *4.1. Training and Evaluation data*

#### *4.1.1. Experimental setup on the RSR dataset (female)*

The training data is drawn from Fisher English (∼120 h) female. This subset of data contains 1.2k utterances with an average duration of 5 minutes per utterance. The choice of Fisher database as a training set was primarily motivated by the requirement of a well-transcribed and standardized data. Following the setup in Dey et al. (2016a); Larcher et al. (2014), the PLDA and JFA models are trained on development set of RSR (female). The Part1 (female) part of RSR data contains 143 female speakers pronouncing 30 fixed passphrases spreading over nine sessions. Speakers are divided into three parts, background, development and evaluation portions. Data is collected from six different mobile devices with an average duration of 3 s. The development data contains 49 speakers with 12'661 utterances. Evaluation data contains enrollment utterances which are recorded from a fixed mobile device while the test data comes from other devices. The number of speakers in the evaluation part is 47 with 8'810 test utterances. All speech files are down-sampled to 8 kHz for compatibility with other datasets used for system

development.

*4.1.2. Experimental setup on the RedDots dataset (male)*

The training data for experiments on the RedDots is drawn from the Fisher male (∼120 h), similar to the above experimental setup. Since no development data was available for the experiments on RedDots, we choose the RSR, male data from Part1. The Part1 portion (male subset) of the RSR dataset is used as the development data with 42'305 utterances from 157 speakers. We evaluated our systems on the Part4 portion of RedDots database (Lee et al., 2015). The evaluation data of this dataset was distributed during the Interspeech 2016 Special session (RedDots-Challenge, 2016). Compared to RSR, the RedDots contains more sessions of recording of speech data from each speaker. The dataset contains 52 sessions per speaker, with one session per week. Thus the challenge of the systems is to compensate for the long term intra-speaker variability (in addition to inter-speaker variability). We evaluated our system only on the male set of the database (Part4 text-dependent task only). The Part4 consists of 35 speakers pronouncing fixed passphrases (which are different from the phrases of the RSR dataset). It contains a total of 5'696 target trials and 5'229'952 impostor trials, out of which conditions 1, 2 and 3 contain 131'002, 99'264 and 4'999'686 impostor trials, respectively. Similar to previous experimental setup, the speech files are downsampled to 8 kHz for compatibility with other datasets.

### *4.2. Feature Extraction and Voice Activity Detection*

MFCC features of 20 dimensions are extracted from 25 ms of frame of speech signal with 10 ms sliding window, appended with the delta and double delta features. Short-time gaussianization is applied to the features using a 3 s sliding window (Pelecanos and Sridharan, 2001). The Hungarian phoneme recognizer is used to detect voice activity by comparing the sum of posteriors over phone classes with the posterior of silence class to classify each frame as speech or non-speech. This is used to mark the start and end points of the speech region in the utterance (Brummer et al., 2010).

### *4.3. i-vector and JFA configurations*

We implemented two gender-dependent UBMs (one male and another female) comprising of 1024 components using the training data. The parameters of i-vector extractors are estimated with the similar training data as used for UBMs. The dimension of extractors is fixed to 400. The parameters of the JFA systems are estimated with speaker-phrase labels using the development data (as mentioned in the Sections 4.1.1 and 4.1.2). The rank of the eigenchannel matrix $\mathbf{U}$ (of Equation 5) is fixed to 50.

### *4.4. HMM/DNN system configurations*

The DNN, usually trained in ASR fashion, is employed to compute the posteriors of the senone units, which is then used in the DNN-based i-vector and JFA systems parameters estimation process. These posteriors are also used as feature streams in DTW systems. Two gender dependent ASR systems are trained for experiments, one male and another female, with their respective training data (as mentioned in the Sections 4.1.1 and 4.1.2).

Table 1: Performance of the DNN and adapted-DNN (female) based ASR system on RSR and Fisher subset in terms of WER(%).

| Systems/Conditions | Fisher | RSR |
|---|---|---|
| DNN | 24.5 | 85.0 |
| adapted-DNN | 28.2 | 17.1 |

We now proceed to describe the ASR setup as used in the paper. Since the parameters of the two ASR systems are the same, we describe the configuration of one system (female) only. The HMM/GMM system (female) uses context-dependent tri-phone states and a total of 1.5k senone states and 12k Gaussians. This system is used to obtain senone alignments to train the DNN model. The DNN is trained with MFCC input features and a context size of 5 frames. It comprises 4 hidden layers with 1.2k sigmoid units per layer. The output of the DNN is represented by softmax function. It is trained with stochastic gradient descent algorithm to minimize the cross-entropy function between the class labels (senone alignments) and the network output. After the convergence of the algorithm, the posterior probabilities of the senone units corresponding to an input speech frame are obtained at the output of the DNN.

*4.4.1. ASR performance*

The conventional hybrid ASR system uses DNN to estimate acoustic posterior probabilities plugged into the ASR decoder by employing LM. The performance of the female ASR system is evaluated on two batches of data, namely, (i) Fisher female subset with 200 utterances and, (ii) Part1,

RSR female subset consisting of 1k utterances. The ASR system employs a CMU dictionary with 42k words and a tri-gram LM for decoding with word LMs (Motlicek et al., 2015). The Word Error Rates (WER) on both the set are presented in Table 1. The WER of the female DNN is 24.5% on the Fisher subset. Poor performance on the RSR subset is possibly due to acoustic mismatch between the RSR and the training dataset (channel, accent mismatch).

In order to cope with large differences in performance of WER, we adapt the DNN with a small amount of data (∼1 h) from RSR database. The adapted-DNN performs roughly equally well in both the databases (row 2 of Table 1) with absolute improvement of ∼68% in terms of WER on the RSR dataset. The DNN and the adapted-DNN (trained on the female portions) are then used for SV experiments on RSR Part1, female evaluation set only.

The performance of the male-DNN is evaluated only on a Fisher male subset (200 utterances). The WER of this DNN is 30.5%. Since no development data is available from RedDots dataset, the adaptation of DNN could not be done.

### *4.5. Online i-vector configurations*

Two online i-vector systems are developed (for male and female) using the training data as described in Sections 4.1.1 and 4.1.2. Since the parameters of both the systems are similar, we describe the configurations of the female system only. The SS, required to estimate online i-vectors, are computed from short segments of speech signal of duration 200 ms. The i-vector extractor is 400 dimensional. To train the speaker-phone PLDA model, the ASR system developed in the previous subsection is used to obtain senone alignments.

The senones are then mapped to one of 43 monophones to get the phone alignment. The PLDA is trained on the online i-vectors by assigning speaker-phone pair labels to each of the speech frames. The Part1 of RSR dataset is used to train the PLDA. There are a total of 2k classes (speaker-phone pairs) in the development set.

## 5. Experimental Results

In this section, we describe the results obtained with various systems described in Sections 2 and 3. We first present the results on the RSR dataset (Part1, female) and then proceed to RedDots (Part1, male). The conventional approaches include the DTW and model-based SV systems (MAP, i-vector and JFA) both employing GMM posteriors. Since it has been consistently reported in literature that MAP technique outperforms other approaches for text-dependent SV task (Kenny et al., 2014a,c), we consider the MAP system to act as the baseline system in both the experiments on RSR and RedDots. In all the experiments involving PLDA, the input vectors to the model are length normalized. For the MAP, JFA and DTW systems, T-norm score normalization is applied (Barras and Gauvain, 2003; Dey et al., 2016a; Kenny et al., 2014c,a). In our experiments involving i-vectors, we observed that dimensionality reduction technique, like LDA, degraded the performance of the speaker recognition system. Thus, we do not report the performance of the systems using LDA transform.

The various systems considered in this paper are as follows:

- $\mathbf{MAP^{GMM}}$: the speaker models are obtained from UBM-GMM by MAP adaptation.

- **MAP$^{\text{HMM}}$**: a universal HMM is built with the training data. The speaker-phrase models are obtained from the universal HMM by MAP adaptation. The test data are scored against the adapted HMM models.

- **Ivec$_{\text{PLDA}}$**: the conventional i-vector system for speaker recognition obtained using GMM or DNN SS, which are referred to as **Ivec$^{\text{GMM}}_{\text{PLDA}}$** or **Ivec$^{\text{DNN}}_{\text{PLDA}}$** respectively. The system with adapted-DNN SS is labeled as **Ivec$^{\text{DNN-adp}}_{\text{PLDA}}$**.

- **JFA**: this system represents Joint Factor Analysis model. The JFA using GMM SS is referred to as **JFA$^{\text{GMM}}$** while the system using DNN and adapted-DNN SS are referred to as **JFA$^{\text{DNN}}$** and **JFA$^{\text{DNN-adp}}$** respectively.

- **DTW**: raw speech features (MFCCs) and posteriograms obtained from the GMM or DNN are compared using the DTW algorithm in this system. The systems with MFCCs, GMM posteriors, DNN and adapted-DNN posteriors are referred to as **DTW-MFCC**, **DTW-post$^{\text{GMM}}$**, **DTW-post$^{\text{DNN}}$** and **DTW-post$^{\text{DNN-adp}}$** respectively.

- **DTW-onIvec**: this system uses i-vector (estimated over short segments) as input to DTW algorithm. The i-vectors are computed using SS either from GMM or DNN, which are referred to as **DTW-onIvec$^{\text{GMM}}$** and **DTW-onIvec $^{\text{DNN}}$** respectively.

- **DTW-onIvec$_{\text{PLDA}}$**: this system uses PLDA projection (as explained in Section 3.1.2) as input to the DTW algorithm. PLDA is trained either with speaker-phone or speaker-phrase as class definition. DTW

Table 2: Performance of the various GMM based baseline systems on RSR dataset in terms of EER(%). The $\mathbf{MAP^{GMM}}$ outperforms other baseline systems in Cond-all.

| No. | Systems/Conditions | Cond1 | Cond2 | Cond3 | Cond-all |
|---|---|---|---|---|---|
| 1 | $\mathbf{MAP^{GMM}}$ | 0.83 | **2.15** | **0.21** | **0.69** |
| 2 | $\mathbf{MAP^{HMM}}$ | **0.71** | 4.42 | 0.42 | 1.32 |
| 3 | $\mathbf{Ivec^{GMM}_{PLDA}}$ | 1.24 | 2.82 | 0.32 | 0.91 |
| 4 | $\mathbf{JFA^{GMM}}$ | 1.42 | 2.34 | 0.41 | 0.71 |

system with PLDA (trained with speaker-phone labels) projection obtained using GMM posteriors (for online i-vector extraction) is referred to as $\mathbf{DTW\text{-}onIvec^{GMM}_{PLDA,\ phn}}$ while with DNN is referred to as $\mathbf{DTW\text{-}onIvec^{DNN}_{PLDA,\ phn}}$. The systems, with PLDA trained using speaker-phrase classes are referred to as $\mathbf{DTW\text{-}onIvec^{GMM}_{PLDA,\ phr}}$ and $\mathbf{DTW\text{-}onIvec^{DNN}_{PLDA,\ phr}}$.

### *5.1. Experiments on the RSR data (female)*

The experiments are conducted with the training and evaluation data as detailed in Section 4.1.1. We first describe the model-based SV systems using GMM and DNN posteriors and then move on to DTW systems.

#### *5.1.1. Model-based SV systems with GMM posteriors*

Table 2 compares the performance of various model-based SV systems exploiting GMM posteriors. It is to be noted that the results presented here are comparable or better than those published in Larcher et al. (2014); Kenny et al. (2014c). The simple MAP technique, $\mathbf{MAP^{GMM}}$ (row 1) achieves the best results among the model-based SV systems, which is consistent with the

results published in the literature. T-norm is applied on $\mathbf{MAP}^{\mathbf{GMM}}$ scores with improvement of 24% relative EER (from 2.85% to 2.15% absolute) for condition 2. The $\mathbf{MAP}^{\mathbf{HMM}}$ performs worse than the $\mathbf{MAP}^{\mathbf{GMM}}$ in Cond-all, however in Cond1, the former system performs better than the latter system due to the ability of the HMM to capture sequential information.

In text-independent SV scenario, the $\mathbf{Ivec}^{\mathbf{GMM}}_{\mathbf{PLDA}}$ system outperforms $\mathbf{MAP}^{\mathbf{GMM}}$ as evident by the success of the technique in past SV evaluations. However, in text-dependent scenario, the $\mathbf{Ivec}^{\mathbf{GMM}}_{\mathbf{PLDA}}$ system (row 2 of Table 2) performs worse, which may be due to the duration of the test utterances.

We explored JFA system as well, as it has shown to be a dominating modeling technique for text-dependent SV scenario. The latent factor ($\mathbf{z}$) of the JFA model (Equation 5), which characterizes the speaker-phrase, is used to compute the cosine distance between the enrollment and test utterances. T-norm is applied to the scores produced by the JFA model. This system ($\mathbf{JFA}^{\mathbf{GMM}}$) performs better than the $\mathbf{Ivec}^{\mathbf{GMM}}_{\mathbf{PLDA}}$ in condition 2 (compare row 3 and row 2), thus showing that the matrix $\mathbf{D}$ is able to model the speaker-phrase characteristics better than the matrix $\mathbf{V}$ of the PLDA model of Equation 4. The JFA system can be built with only the development data of RSR database without the need of any Fisher database.

*5.1.2. Model-based SV systems with DNN posteriors*

As explained in Section 2, the $\mathbf{Ivec}_{\mathbf{PLDA}}$ and **JFA** systems benefit by incorporating linguistic information from HMM/DNN. The DNN acoustic model is employed to estimate the senone posteriors, which is then subsequently fed to i-vector extraction process. The 10 top scoring DNN posteriors are used to estimate the parameters of the i-vector and JFA models as given

Table 3: Performance of the various DNN-based SV systems on RSR dataset in terms of EER(%). The JFA system is the best performing system.

| No. | Systems/Conditions | Cond1 | Cond2 | Cond3 | Cond-all |
|---|---|---|---|---|---|
| 1 | $\mathbf{Ivec}^{\mathbf{DNN}}_{\mathbf{PLDA}}$ | 0.71 | 2.52 | 0.21 | 0.73 |
| 2 | $\mathbf{JFA}^{\mathbf{DNN}}$ | **0.12** | **0.84** | **0.02** | **0.21** |

by Equations 3 and 5 respectively. The back-end classifier of the i-vector model (PLDA) is trained with multiple instances of speaker-phrase classes (from development data).

Table 3 shows the performance of the model-based SV systems with DNN posteriors. We observe that integrating DNN posteriors in the $\mathbf{Ivec}_{\mathbf{PLDA}}$ and **JFA** systems consistently improves the performance (compare rows 1, 2 of Table 3 with rows 2, 3 of Table 2). In particular, $\mathbf{Ivec}^{\mathbf{DNN}}_{\mathbf{PLDA}}$ improves upon $\mathbf{Ivec}^{\mathbf{GMM}}_{\mathbf{PLDA}}$ by 22% relative EER (from 0.91% to 0.73% absolute) for Cond-all condition. The $\mathbf{JFA}^{\mathbf{DNN}}$ achieves good results and clearly outperforms the $\mathbf{JFA}^{\mathbf{GMM}}$, this system performs better than the $\mathbf{MAP}^{\mathbf{GMM}}$ across all conditions by 66% relative EER (from 0.69% vs 0.21% absolute) for Cond-all. This validates the hypothesis that linguistic units of the speech signal are important for the i-vector and JFA SV approaches.

*5.1.3. DTW systems*

The **DTW-MFCC** technique has been explored for text-dependent SV task in the past. It assumes that MFCCs contain speaker and content discriminating information, to be exploited by DTW algorithm. Furthermore, we experimented with GMM and ($\mathbf{DTW\text{-}post}^{\mathbf{GMM}}$), DNN posteriors

Table 4: Performance of the various DTW systems on RSR dataset in terms of EER(%). The DTW system using DNN posterior features performs better in content-mismatch conditions.

| No. | Systems/Conditions | Cond1 | Cond2 | Cond3 | Cond-all |
|---|---|---|---|---|---|
| 1 | **DTW-MFCC** | 0.38 | 4.52 | 0.11 | 1.23 |
| 2 | **DTW-post$^{\mathbf{GMM}}$** | 0.13 | **4.51** | 0.11 | 1.22 |
| 3 | **DTW-post$^{\mathbf{DNN}}$** | **0.04** | 4.61 | **0.02** | **1.05** |

(**DTW-post$^{\mathbf{DNN}}$**) constituting input to DTW. It can be observed from Table 4 that all the DTW techniques achieve better results than the baseline model-based SV systems (**MAP$^{\mathbf{GMM}}$**, **Ivec$^{\mathbf{GMM}}_{\mathbf{PLDA}}$** and **JFA$^{\mathbf{GMM}}$** of Table 2) for content-mismatch conditions. However, for condition 2, the performance is significantly worse than the model-based SV systems with GMM posteriors (Table 2). It can be observed from Table 4 that **DTW-post$^{\mathbf{DNN}}$** (row 3) outperforms the **MAP$^{\mathbf{GMM}}$** for conditions 1 and 3 by 95% relative EER (from 0.83% vs 0.04% absolute) and 90% relative EER(from 0.21% vs 0.02% absolute) respectively.

*5.1.4. Systems using Adapted-DNN*

Table 5 shows the performance of various systems (i-vector, JFA and DTW) exploiting posteriors obtained at the output of adapted-DNN. The main motivation of adaptation is to obtain better alignment of the evaluation data. The **Ivec$^{\mathbf{DNN\text{-}adp}}_{\mathbf{PLDA}}$** performs better than **Ivec$^{\mathbf{DNN}}_{\mathbf{PLDA}}$** (compare row 1 of Table 3 and row 1 of Table 5) across all conditions. This system performs better than the **MAP$^{\mathbf{GMM}}$** by 26% relative EER (from 0.69% to 0.52%

Table 5: Performance of the various adapted-DNN based systems on RSR dataset in terms of EER (%). The JFA system is the best performing system.

| No. | Systems/Conditions | Cond1 | Cond2 | Cond3 | Cond-all |
|---|---|---|---|---|---|
| 1 | $\mathbf{Ivec}_{\mathbf{PLDA}}^{\mathbf{DNN\text{-}adp}}$ | 0.15 | 2.17 | 0.02 | 0.52 |
| 2 | $\mathbf{JFA}^{\mathbf{DNN\text{-}adp}}$ | 0.11 | **0.71** | 0.02 | **0.21** |
| 3 | $\mathbf{DTW\text{-}post}^{\mathbf{DNN\text{-}adp}}$ | **0.02** | 14.52 | **0.01** | 2.61 |

absolute) for Cond-all.

The senone posteriors of the adapted-DNN are used to estimate the parameters of the JFA model as given by Equation 5 (matrices **D** and **U**) and subsequently the latent variable **z** (during enrollment and testing phase). From Table 5 we observe that $\mathbf{JFA}^{\mathbf{DNN\text{-}adp}}$ further improves upon $\mathbf{JFA}^{\mathbf{DNN}}$ (compare row 2 of Table 3 and row 2 of Table 5), particularly for Cond2, indicating that the DNN adaptation is useful in the i-vector and JFA systems.

The senone posteriors from the adapted-DNN are used as features for the DTW algorithm. We observe that $\mathbf{DTW\text{-}post}^{\mathbf{DNN\text{-}adp}}$ performs better than $\mathbf{Ivec}_{\mathbf{PLDA}}^{\mathbf{DNN\text{-}adp}}$ and $\mathbf{JFA}^{\mathbf{DNN\text{-}adp}}$ for content-mismatch conditions while significantly degrading performance for condition 2. This degradation in performance is due to the content-discriminating features. We attempt to solve this problem by extracting speaker-discriminating features for DTW algorithm.

*5.1.5. DTW systems with online i-vectors*

The **DTW-onIvec** extracts i-vectors on short segments (online i-vectors), which are then used as input features to DTW algorithm. It can be observed

Table 6: Performance of the various DTW systems using online i-vector features on RSR database in terms of EER(%). The **DTW-onIvec$^{\mathbf{DNN}}_{\mathbf{PLDA,\ phn}}$** is the best performing system.

| No. | Systems/Conditions | Cond1 | Cond2 | Cond3 | Cond-all |
|---|---|---|---|---|---|
| 1 | **DTW-onIvec$^{\mathbf{GMM}}$** | 0.21 | 1.52 | 0.05 | 0.45 |
| 2 | **DTW-onIvec$^{\mathbf{DNN}}$** | 0.03 | 0.75 | 0.02 | 0.23 |
| 3 | **onIvec$^{\mathbf{GMM}}_{\mathbf{PLDA}}$** | 4.41 | 6.49 | 1.03 | 1.93 |
| 4 | **onIvec$^{\mathbf{DNN}}_{\mathbf{PLDA}}$** | 1.62 | 4.42 | 0.39 | 1.06 |
| 5 | **DTW-onIvec$^{\mathbf{GMM}}_{\mathbf{PLDA,\ phn}}$** | 0.15 | 1.21 | 0.02 | 0.35 |
| 6 | **DTW-onIvec$^{\mathbf{DNN}}_{\mathbf{PLDA,\ phn}}$** | **0.02** | **0.65** | **0.01** | **0.18** |
| 7 | **DTW-onIvec$^{\mathbf{DNN}}_{\mathbf{PLDA,\ phr}}$** | 0.05 | 0.86 | 0.03 | 0.24 |

from Table 6 that the **DTW-onIvec$^{\mathbf{GMM}}$** and **DTW-onIvec$^{\mathbf{DNN}}$** outperform the baseline **MAP$^{\mathbf{GMM}}$** by about 35% relative EER (from 0.69% to 0.45% absolute) and 67% relative EER (from 0.69% to 0.23% absolute) for Cond-all condition. This indicates that online i-vectors represent speakers sufficiently well. The DTW algorithm plays an important role in achieving good performance by the **DTW-onIvec** system. Therefore, without the sequence matching capability (of the DTW algorithm), the online i-vector system performing an averaging operation instead of preserving the sequential information is expected to provide worse results than **DTW-onIvec**. To test this hypothesis, we conducted an experiment by building a system (similar to **Ivec$_{\mathbf{PLDA}}$**) as follows. A sequence of online i-vectors is extracted which is then averaged to obtain a representative i-vector of the utterance. The PLDA is trained using these averaged online i-vectors as features assum-

ing speaker-phrase as classes. The distance between the enrollment and test speech signal is computed using the PLDA model with the averaged online i-vectors. We built two systems applying this strategy, one with GMM posteriors and another with DNN posteriors, which are referred to as $\mathbf{onIvec}^{\mathbf{GMM}}_{\mathbf{PLDA}}$ and $\mathbf{onIvec}^{\mathbf{DNN}}_{\mathbf{PLDA}}$ respectively in Table 6. We observe that $\mathbf{onIvec}^{\mathbf{GMM}}_{\mathbf{PLDA}}$ and $\mathbf{onIvec}^{\mathbf{DNN}}_{\mathbf{PLDA}}$ performs worse than **DTW-onIvec** (compare rows 3, 4 vs rows 1, 2 of Table 6). This result highlights the significance of DTW algorithm, in addition to the online i-vectors, in obtaining low error rates.

From Table 6, it can be observed that applying PLDA on top of the online i-vector features further improves the performance. The $\mathbf{DTW\text{-}onIvec}^{\mathbf{DNN}}_{\mathbf{PLDA},\ \mathbf{phn}}$ improves over the $\mathbf{MAP}^{\mathbf{GMM}}$ baseline system by 74% relative EER for Cond-all. In Section 3, we discussed the two possible methods of defining classes in the PLDA model with online i-vector features, which are speaker-phrase and speaker-phone. We observe that both the systems, $\mathbf{DTW\text{-}onIvec}^{\mathbf{DNN}}_{\mathbf{PLDA},\ \mathbf{phn}}$ and $\mathbf{DTW\text{-}onIvec}^{\mathbf{DNN}}_{\mathbf{PLDA},\ \mathbf{phr}}$, perform similar for all conditions. We did not obtain better results of **DTW-onIvec** using adapted-DNN than DNN and thus we are not presenting the results.

*5.1.6. Summary of experiments on RSR database*

The minDCF and DET plot of some of the best performing systems are presented in Table 7 and Figure 2 respectively for Cond-all condition only. These systems include, (i) the $\mathbf{MAP}^{\mathbf{GMM}}$ baseline, (ii) $\mathbf{Ivec}^{\mathbf{DNN\text{-}adp}}_{\mathbf{PLDA}}$, (iii) $\mathbf{JFA}^{\mathbf{DNN\text{-}adp}}$ and, (iv) $\mathbf{DTW\text{-}onIvec}^{\mathbf{DNN}}_{\mathbf{PLDA},\ \mathbf{phn}}$. It is to be noted that $\mathbf{DTW\text{-}onIvec}^{\mathbf{DNN}}_{\mathbf{PLDA},\ \mathbf{phn}}$ improves by 71% relative minDCF (from 0.329% to 0.094% absolute) compared to the baseline $\mathbf{MAP}^{\mathbf{GMM}}$.

Table 7: Performance of the various systems on RSR database in terms of EER(%)/minDCF(%) in Cond-all condition.

| No. | Systems/Conditions | Posteriors | Cond-all |
|---|---|---|---|
| 1 | $\mathbf{MAP^{GMM}}$ (Table 2) | GMM | 0.69/0.329 |
| 2 | $\mathbf{JFA^{DNN\text{-}adp}}$ (Table 5) | DNN | 0.21/0.129 |
| 3 | $\mathbf{Ivec^{DNN\text{-}adp}_{PLDA}}$ (Table 5) | DNN | 0.51/0.339 |
| 4 | $\mathbf{DTW\text{-}onIvec^{DNN}_{PLDA,\ phn}}$ (Table 6) | DNN | **0.18/0.094** |

*5.2. Experiments on the RedDots database (male)*

Table 8 compares the performance of all systems on RedDots dataset across all the conditions. We consider the MAP system ($\mathbf{MAP^{GMM}}$) using GMM posterior as the baseline since it has shown to provide good performance in Zeinali et al. (2016). The model-based SV systems perform worse on the RedDots database compared to RSR database (Dey et al., 2016a). As it has been observed from the experiments on RSR database, the model-based SV approaches with DNN acoustic model outperform those employing GMM. Thus, only the results of DNN based i-vector and JFA systems are reported on the RedDots database.

From Table 8, it can be observed that $\mathbf{MAP^{GMM}}$ provides EER of 1.23% for Cond-all. The performance of the MAP system is worse on the RedDots than on the RSR database across all conditions, possibly due to long-term intra-speaker variability. The $\mathbf{MAP^{HMM}}$ outperforms $\mathbf{MAP^{GMM}}$ on this part of the database by 26% relative EER (from 1.23% to 0.94% absolute) on Cond-all.

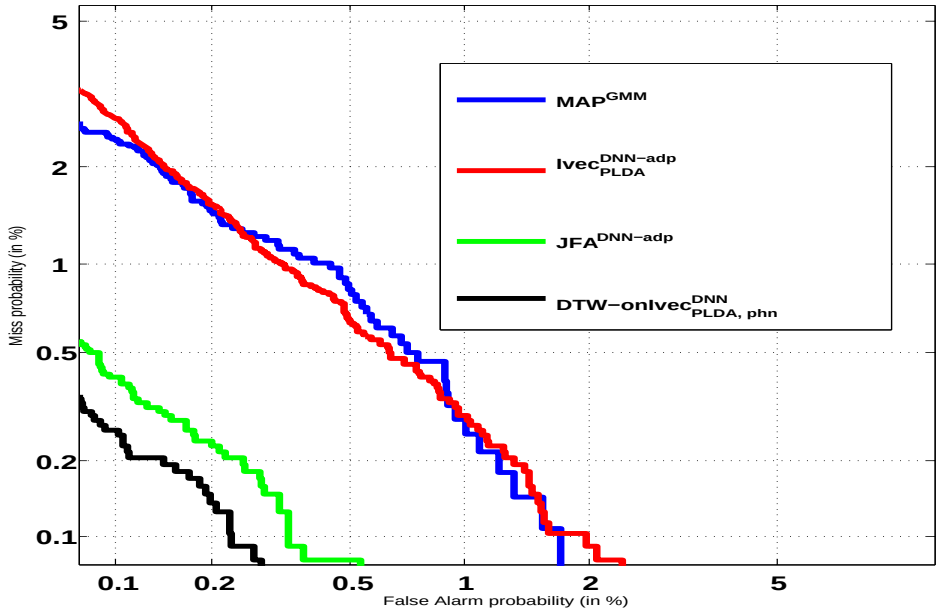


Figure 2: DET curve of the systems presented in Table 7 on RSR database.

The $\mathbf{Ivec}^{\mathbf{DNN}}_{\mathbf{PLDA}}$ and $\mathbf{JFA}^{\mathbf{DNN}}$ systems do not achieve good results as compared to $\mathbf{MAP}^{\mathbf{GMM}}$. The poor performance of i-vector and JFA systems can be possibly attributed to the fact that factoring out the content-variability with speaker-phrase data from RSR is not a good choice.

The $\mathbf{DTW\text{-}post}^{\mathbf{DNN}}$ (row 5 of Table 8) performs better than model-based SV systems in content-mismatch trials (conditions 1 and 3) as it explicitly matches the content. In speaker-mismatch trials, even the $\mathbf{DTW\text{-}post}^{\mathbf{GMM}}$ (row 6) performs better than $\mathbf{DTW\text{-}post}^{\mathbf{DNN}}$.

The $\mathbf{DTW\text{-}onIvec}^{\mathbf{DNN}}$ performs better than $\mathbf{MAP}^{\mathbf{GMM}}$ by 55% relative EER (from 1.23% to 0.55% absolute) for Cond-all. Thus, on this database as well, the online i-vector representation with DTW algorithm achieves better results than $\mathbf{Ivec}^{\mathbf{DNN}}_{\mathbf{PLDA}}$, $\mathbf{JFA}^{\mathbf{DNN}}$ and $\mathbf{MAP}^{\mathbf{GMM}}$. We experimented with using

Table 8: Performance of all the systems on RedDots (Part4) database in terms of EER(%). The Cond-all refers to the system performance across all the 3 conditions.

| No. | Systems/Conditions | Cond1 | Cond2 | Cond3 | Cond-all |
|---|---|---|---|---|---|
| 1 | $\mathbf{MAP^{GMM}}$ | 5.62 | 4.04 | 0.90 | 1.23 |
| 2 | $\mathbf{MAP^{HMM}}$ | 2.63 | 3.72 | 0.73 | 0.94 |
| 3 | $\mathbf{Ivec^{DNN}_{PLDA}}$ | 6.10 | 3.03 | 0.97 | 1.29 |
| 4 | $\mathbf{JFA^{DNN}}$ | 7.21 | 4.43 | 1.34 | 1.85 |
| 5 | $\mathbf{DTW\text{-}post^{DNN}}$ | **0.62** | 7.62 | 0.54 | 1.13 |
| 6 | $\mathbf{DTW\text{-}post^{GMM}}$ | 0.89 | 4.92 | 0.76 | 0.96 |
| 7 | $\mathbf{DTW\text{-}onIvec^{DNN}}$ | 0.99 | 2.69 | 0.44 | **0.55** |
| 8 | $\mathbf{DTW\text{-}onIvec^{DNN}_{PLDA,\ phn}}$ | 0.81 | **2.61** | **0.38** | **0.55** |
| 9 | $\mathbf{DTW\text{-}onIvec^{DNN}_{PLDA,\ phr}}$ | 1.24 | 2.85 | 0.51 | 0.62 |

PLDA on top of online i-vectors. We observe that $\mathbf{DTW\text{-}onIvec^{DNN}_{PLDA,\ phn}}$ further improves upon $\mathbf{DTW\text{-}onIvec^{DNN}}$ with improvement of 3% relative EER (from 2.69% to 2.61% absolute) for Cond2. However, it can also be observed from Table 8 that training the PLDA with speaker-phrase labels degrades the performance. An explanation of the performance degradation is possibly due to training PLDA with speaker-phrase classes from RSR dataset (which do not match the evaluation phrases of RedDots).

## 6. Conclusions

In this paper, we presented model- (MAP, i-vector and JFA) and DTW-based techniques for performing text-dependent SV with fixed phrases. We validated the techniques on two databases, female part of RSR and male part

of RedDots. We experimented with model-based SV systems using GMM and DNN posteriors. From results, we observed that MAP technique performs the best among the model-based SV approaches exploiting GMM posteriors. Integrating DNN posteriors in the i-vector and JFA systems achieves good results across all the conditions, with JFA improves upon the MAP technique by 66% relative EER for Cond-all in RSR dataset. This gain in performance is consistent with the results published for text-dependent and text-independent SV scenarios. Additional gain in performance is obtained with adapted-DNN, more particularly by the JFA technique. It clearly shows that obtaining better alignment for the evaluation data results in better performance.

The DTW algorithm offers an easy method to match the sequential patterns of the train and test templates. Being a non-parametric method, it does not require any training data for the development. We experimented with different input features for the DTW algorithm, namely MFCCs, GMM and DNN posteriors. In content-mismatch conditions, the DTW systems provide better results than the model-based SV systems. In particular, the DTW algorithm using DNN posteriors outperforms the MAP system in condition 1 by 95% relative EER in RSR dataset.

However, DTW system using DNN posteriors performs worse than MAP technique in speaker-mismatch condition. This degradation in performance is due to content-discriminating features. In this paper, we address this problem by extracting speaker specific information by employing i-vector system. We extract online i-vectors (for short segments) using the i-vector extractor of the speech utterance resulting in sequences of online i-vectors

extracted from enrollment and test utterances. The DTW algorithm is then used to match the train and test templates of online i-vectors. We found that this approach outperforms the MAP based system by 67% relative EER for Over-all condition in RSR database.

The PLDA is usually applied in state-of-the-art SV systems as a channel compensation model. In this paper, we experimented with two different definition of class labels, namely, (i) speaker-phrase, and (ii) speaker-phone for training the PLDA. Although on RSR database, we obtained similar performance with both the strategies for defining classes, but on RedDots we obtained considerable performance benefit with speaker-phone labels.

## References

N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, P. Ouellet, Front-End Factor Analysis for Speaker Verification, IEEE Transactions on Audio, Speech, and Language Processing 19 (4) (2011) 788–798.

P. Motlicek, S. Dey, S. Madikeri, L. Burget, Employment of subspace Gaussian mixture models in speaker recognition, in: International Conference on Acoustics, Speech and Signal Processing, 4445–4449, 2015.

P. Kenny, T. Stafylakis, J. Alam, P. Ouellet, M. Kockmann, Joint Factor Analysis for Text-Dependent Speaker Verification, in: Proceedings of Odyssey, 1–8, 2014a.

S. Dey, S. Madikeri, M. Ferras, P. Motlicek, Deep Neural Network based Posteriors for Text-dependent Speaker Verification, in: International Conference on Acoustics, Speech and Signal Processing, 5050–5054, 2016a.

P. Kenny, T. Stafylakis, P. Ouellet, M. Alam, JFA-based front ends for speaker recognition, in: International Conference on Acoustics, Speech and Signal Processing, 1705–1709, 2014b.

S. Novoselov, T. Pekhovsky, A. Shulipa, A. Sholokhov, Text-dependent GMM-JFA system for password based speaker verification, in: International Conference on Acoustics, Speech and Signal Processing, 729–737, 2014.

H. Zeinali, E. Kalantari, H. Sameti, H. Hadian, Telephony text-prompted speaker verification using i-vector representation, in: International Conference on Acoustics, Speech and Signal Processing, 4839–4843, 2015.

S. Zhang, M. Mak, H. Meng, Speaker verification via high-level feature based phonetic-class pronunciation modeling, IEEE Transactions on Computers 56 (9) (2007) 1189–1198.

Y. Lei, N. Scheffer, L. Ferrer, M. McLaren, A novel scheme for speaker recognition using a phonetically-aware deep neural network, in: International Conference on Acoustics, Speech and Signal Processing, 1695–1699, 2014.

A. Larcher, K. Lee, B. Ma, H. Li, Text-dependent speaker verification: Classifiers, databases and RSR2015, Speech Communication 60 (2014) 56–77.

H. Su, S. Wegmann, Factor analysis based speaker verification using ASR, in: Interspeech, 2223–2227, 2016.

S. Jelil, R. K. Das, R. Sinha, S. M. Prasanna, Speaker Verification Using Gaussian Posteriorgrams on Fixed Phrase Short Utterances, in: Interspeech, 1042–1046, 2015.

S. Madikeri, I. Himawan, P. Motlicek, M. Ferras, Integrating Online I-vector extractor with Information Bottleneck based Speaker Diarization system, in: Interspeech, 3105–3109, 2015.

K. A. Lee, A. Larcher, G. Wang, P. Kenny, N. Brümmer, D. Van Leeuwen, H. Aronowitz, M. Kockmann, C. Vaquero, B. Ma, et al., The RedDots Data Collection for Speaker Recognition, in: Interspeech, 2996–3000, 2015.

D. A. Reynolds, T. F. Quatieri, R. B. Dunn, Speaker Verification Using Adapted Gaussian Mixture Models, Digital Signal Processing 10 (1) (2000) 19–41.

O. Glembek, L. Burget, P. Matějka, M. Karafiát, P. Kenny, Simplification and Optimization of i-vector Extraction, in: International Conference on Acoustics, Speech and Signal Processing, 4516–4519, 2011.

D. G. Romero, C. Y. E. Wilson, Analysis of ivector Length Normalization in Speaker Recognition Systems, in: Interspeech, 249–252, 2011.

P. Kenny, T. Stafylakis, P. Ouellet, M. J. Alam, JFA-based front ends for speaker recognition, in: International Conference on Acoustics, Speech and Signal Processing, 1705–1709, 2014c.

P. Kenny, G. Boulianne, P. Dumouchel, Eigenvoice modeling with sparse training data, IEEE Transactions on Audio, Speech, and Language Processing 13 (3) (2005) 335–354.

R. J. Vogt, B. J. Baker, S. Sridharan, Modelling session variability in text independent speaker verification, in: Interspeech, 3117–3120, 2005.

R. Vogt, S. Sridharan, Explicit modelling of session variability for speaker verification, Computer Speech & Language 22 (1) (2008) 17–38.

A. Park, T. J. Hazen, ASR dependent techniques for speaker identification., in: Interspeech, 1337–1340, 2002.

D. E. Sturim, D. A. Reynolds, R. B. Dunn, T. F. Quatieri, Speaker verification using text-constrained Gaussian mixture models, in: International Conference on Acoustics, Speech and Signal Processing, 677–680, 2002.

B. J. Baker, R. J. Vogt, S. Sridharan, Gaussian mixture modelling of broad phonetic and syllabic events for text-independent speaker verification, in: European Conference on Speech Communication and Technology, 2429–2432, 2005.

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

Y. Huang, M. Slaney, M. L. Seltzer, Y. Gong, Towards better performance with heterogeneous training data in acoustic modeling using deep neural networks., in: Interspeech, 845–849, 2014.

R. Gemello, F. Mana, S. Scanzio, P. Laface, R. De Mori, Adaptation of hybrid ANN/HMM models using linear hidden transformations and conservative training, in: International Conference on Acoustics, Speech and Signal Processing, 1189–1192, 2006.

B. Li, K. C. Sim, Comparison of discriminative input and output transformations for speaker adaptation in the hybrid NN/HMM systems, in: Interspeech, 526–529, 2010.

K. Amino, T. Sugawara, T. Arai, Idiosyncrasy of nasal sounds in human speaker identification and their acoustic properties, Acoustical science and technology 27 (4) (2006) 233–235.

E. Shriberg, Higher-level features in speaker recognition, in: Speaker Classification I, Springer, 241–259, 2007.

W. M. Campbell, J. P. Campbell, D. A. Reynolds, D. A. Jones, T. R. Leek, Phonetic speaker recognition with support vector machines, in: Advances in neural information processing systems, 2003.

L. J. Rodriguez-Fuentes, A. Varona, M. Penagarikano, G. Bordel, M. Diez, High-performance Query-by-Example Spoken Term Detection on the SWS 2013 evaluation, in: International Conference on Acoustics, Speech and Signal Processing, 7819–7823, 2014.

E. Keogh, C. A. Ratanamahatana, Exact indexing of dynamic time warping, Knowledge and information systems 7 (3) (2005) 358–386.

M. Versteegh, R. Thiolliere, T. Schatz, X. N. Cao, X. Anguera, A. Jansen, E. Dupoux, The zero resource speech challenge 2015, in: Interspeech, 2015.

M. Brown, L. Rabiner, An adaptive, ordered, graph search technique for dynamic time warping for isolated word recognition, IEEE Transactions on Audio, Speech, and Language Processing 30 (4) (1982) 535–544.

V. Ramasubramanian, A. Das, V. P. Kumar, Text-Dependent Speaker-Recognition Using One-Pass Dynamic Programming Algorithm, in: International Conference on Acoustics, Speech and Signal Processing, 901–904, 2006.

V. Peddinti, G. Chen, D. Povey, S. Khudanpur, Reverberation robust acoustic modeling using i-vectors with time delay neural networks, in: Interspeech, 2015.

D. Vijayasenan, F. Valente, H. Bourlard, An Information Theoretic Combination of MFCC and TDOA Features for Speaker Diarization, IEEE Transactions on Audio, Speech, and Language Processing 19 (2) (2011) 431–438.

N. Tishby, F. C. Pereira, W. Bialek, The information bottleneck method, arXiv preprint physics/0004057 .

D. G. Romero, A. McCree, Supervised domain adaptation for I-vector based speaker recognition, in: International Conference on Acoustics, Speech and Signal Processing, 4047–4051, 2014.

S. Dey, S. Madikeri, P. Motlicek, Information theoretic clustering for unsupervised domain-adaptation, in: International Conference on Acoustics, Speech and Signal Processing, 5580–5584, 2016b.

RedDots-Challenge, URL `https://sites.google.com/site/thereddotsproject/`, 2016.

J. Pelecanos, S. Sridharan, Feature Warping for Robust Speaker Verification, in: Proceedings of Odyssey, 213–218, 2001.

N. Brummer, L. Burget, P. Kenny, P. Matejka, E. de Villiers, M. Karafiat, M. Kockmann, O. Glembek, O. Plchot, D. Baum, et al., ABC system description for NIST SRE 2010, Proc. NIST 2010 Speaker Recognition Evaluation (2010) 1–20.

C. Barras, J.-L. Gauvain, Feature and score normalization for speaker verification of cellular data, in: International Conference on Acoustics, Speech and Signal Processing, 46–49, 2003.

H. Zeinali, H. Sameti, L. Burget, J. Černockỳ, N. Maghsoodi, P. Matějka, i-vector/HMM Based Text-dependent Speaker Verification System for RedDots Challenge, in: Interspeech, 440–444, 2016.