LIKELIHOOD-RATIO EMPIRICAL KERNELS FOR I-VECTOR BASED PLDA-SVM SCORING

Man-Wai Mak and Wei Rao

Dept. of Electronic and Information Engineering
The Hong Kong Polytechnic University, Hong Kong
Email: enmwmak@polyu.edu.hk

ABSTRACT

Likelihood ratio (LR) scoring in PLDA speaker verification systems only uses the information of background speakers *implicitly*. This paper exploits the notion of empirical kernel maps to incorporate background speaker information into the scoring process *explicitly*. This is achieved by training a scoring SVM for each target speaker based on a kernel in the empirical feature space. More specially, given a test i-vector and the identity of the target under test, a score vector is constructed by computing the LR scores of the test i-vector with respect to the target-speaker's i-vectors and a set of background-speakers' i-vectors. While in most situations, only one target-speaker i-vector is available for training the SVM, this paper demonstrates that if the enrollment utterance is sufficiently long, a number of target-speaker i-vectors can be generated by an utterance partitioning and resampling technique, resulting in much better scoring SVMs. Results on NIST 2010 SRE suggests that the idea of incorporating background speaker information into PLDA scoring through training speaker-dependent SVMs together with the utterance partitioning techniques can boost the performance of i-vector based PLDA systems significantly.

Index Terms— I-vectors; probabilistic linear discriminant analysis; empirical kernel maps; likelihood ratio kernels

1. INTRODUCTION

1.1. Motivation of Work

The i-vector approach [1] to speaker verification is based on the idea of joint factor analysis (JFA) [2, 3] in which the channel and speaker spaces are considered as a single space called the total variability space. Recent research has been focusing on using heavy-tailed probabilistic linear discriminant analysis (PLDA) [4] and Gaussian PLDA [5, 6] to suppress session variability in i-vectors [1]. In these systems, given a test i-vector and a target-speaker i-vector, the verification decision is based on the log-likelihood ratio (LR) score derived from two hypotheses: (1) the test i-vector and the target-speaker i-vector are from the same speaker and (2) these two i-vectors are from two different speakers. Because the computation of the likelihood ratio does not involve other i-vectors, this scoring method *implicitly* uses background information through the universal background model (UBM) [7] and the total variability matrix when estimating the i-vectors and through the PLDA loading matrix when computing the LR score. This LR scoring method is computationally efficient, and because it is a Bayesian approach, score normalization is not necessary. However, the implicit use of background information is a drawback of this method.

This paper explores the possibility of using discriminative model for scoring so that the use of background information becomes *explicit*. Specifically, for each target speaker, an empirical score space with dimension equal to the number of training i-vectors for this target speaker is defined by using the idea of empirical kernel maps [8–10]. Given an i-vector, a score vector living in this space is formed by computing the LR score of this i-vector with respect to each of the training i-vectors. A speaker-dependent support vector machine (SVM) – referred to as empirical LR SVM – can then be trained using the training score vectors. During verification, given a test i-vector and the target-speaker under test, the LR score is mapped to a score vector, which is then fed to the target-speaker's SVM to obtain the final test score.

Using speaker-dependent SVMs for i-vector scoring is not very popular in speaker verification, primary because of its inferior performance when compared with cosine distance scoring [1] and PLDA scoring [4]. The poorer performance of SVM scoring, however, is mainly due to the severe imbalance between the number of target-speaker vectors and the number of background speaker vectors. Typically, there is only one i-vector per target speaker, because only one enrollment session is available. This difficulty, however, can be overcome by a technique called utterance partitioning with acoustic vector resampling (UP-AVR) [11–13]. This technique has been successfully applied to both GMM-SVM [14–16] and i-vector based systems [17].

The idea of UP-AVR is based on the observation that the discriminative power of i-vectors reaches a plateau quickly when the utterance length increases [13]. This means that the speaker-dependent information in a long utterance cannot be fully utilized if only one i-vector is extracted from the utterance. To maximize the utilization, UP-AVR first reshuffles the acoustic vector sequence of a long utterance; then the reshuffled acoustic-sequence is partitioned into equal-length segments, with each segment independently used for estimating an i-vector. This frame-index randomization and partitioning process can be repeated several times to produce a desirable number of i-vectors for each conversation. It has been demonstrated in [17] that increasing the number of target-speaker i-vectors can help the SVM training algorithm to find better decision boundaries, thus making SVM scoring outperforms cosine-distance scoring. In this paper, we further demonstrate that UP-AVR is indispensable for training the speaker-dependent empirical LR SVMs.

1.2. Related Works

There has been previous work that uses discriminative models for PLDA scoring. For example, in [18, 19], for each verification trial,

This work was in part supported by The Hong Kong Research Grant Council Grant No. PolyU5264/09E and HKPolyU Grant No. G-YJ86.

the LR score of a test i-vector and a target-speaker i-vector is expressed as a dot product between a speaker-independent weight vector and a vector whose elements are derived from these two i-vectors in the trial. The weight vector is discriminatively trained by logistic regression or SVM training algorithm using all of the available i-vector pairs (same-speaker pairs and different-speaker pairs) in the development set. Essentially, this method trains a binary classify that takes a pair of i-vectors as input and produces a score that better reflects the similarity/difference of the pair. This idea has been extended to gender-independent PLDA scoring in [20].

The SVM scoring method proposed in this paper is different from these previous studies in three aspects. First, all of these studies use a large number of same-speaker and different-speaker i-vector pairs to train a speaker-independent SVM for scoring. As a result, the discrimination between the same-speaker and different-speaker pairs are encoded in the SVM weights. On the other hand, the proposed method captures the discrimination between the target-speaker and impostors in the SVM weights as well as in the score vectors that live in the empirical feature space. Second, in the proposed method, the SVMs can be optimized for individual target-speakers, whereas the speaker-independent SVM in [18–20] is optimized for all target speakers. Third, because the dimension of the empirical feature space depends on the number of background-speaker i-vectors, it is possible to limit the dimension so that more flexible non-linear SVMs can be applied to the score vectors.

The empirical kernel map in this paper is related to the anchor models [21–23]. However, in the anchor model, a test utterance is projected into a space represented by a set of reference speakers *unrelated* to the target-speakers, whereas the empirical feature space is represented by the target speaker and a set of background speakers.

2. EMPIRICAL LR KERNELS FOR SVMS

2.1. Gaussian PLDA

The aim of LDA [24] is to find a set of orthogonal axes for minimizing the within-class variation and maximizing the between-class separation. In 2007, Prince and Elder [6] proposed a probabilistic approach to the same problem and named the method probabilistic LDA. Kenny [4] applied a similar spirit but replaced the Gaussian distribution of the i-vectors with Student's t-distribution and used a fully Bayesian approach to estimating the model parameters. The resulting model is commonly referred to as heavy-tailed PLDA in the literature. Recently, Garcia-Romero and Espy-Wilson [5] showed that transforming the heavy-tailed distributed i-vectors by whitening and length normalization enables the use of Gaussian assumptions for the PLDA model. It was found that the resulting model, namely Gaussian PLDA, can achieve performance equivalent to that of more complicated systems based on the heavy-tailed assumption. In this paper, we focus on Gaussian PLDA.

Given a set of D-dim length-normalized [5] i-vectors $\mathcal{X} = \{\mathbf{x}_{ij}; i = 1, \ldots, N; j = 1, \ldots, H_i\}$ obtained from N training speakers each has H_i i-vectors, we aim to estimate the latent variables $\mathcal{Z} = \{\mathbf{z}_i;\ i = 1, \ldots, N\}$ and parameters $\boldsymbol{\omega} = \{\boldsymbol{\mu}, \mathbf{W}, \boldsymbol{\Sigma}\}$ of a factor analyzer [6]:

$$\begin{aligned} &\mathbf{x}_{ij} = \boldsymbol{\mu} + \mathbf{W}\mathbf{z}_i + \boldsymbol{\epsilon}_{ij}, \\ &\mathbf{x}_{ij}, \boldsymbol{\mu} \in \Re^D,\ \mathbf{W} \in \Re^{D\times M},\ \mathbf{z}_i \in \Re^M,\ \boldsymbol{\epsilon}_{ij} \in \Re^D, \end{aligned} \tag{1}$$

where $\mathbf{W}$ is a $D \times M$ factor loading matrix ($M < D$), $\boldsymbol{\mu}$ is the global mean of $\mathcal{X}$, $\mathbf{z}_i$'s are the speaker factors, and ϵ_{ij}'s are residual noise assumed to follow a Gaussian distribution with zero mean and covariance $\boldsymbol{\Sigma}$. Because the i-vector dimension D is sufficiently small, it is possible to estimate the full covariance matrix [4, 5]. We used full covariance for $\boldsymbol{\Sigma}$ for all systems.

Given a test i-vector $\mathbf{x}_t$ and target-speaker's i-vector $\mathbf{x}_s$, the likelihood ratio score can be computed as follows:

$$\begin{aligned} S_{\text{LR}}(\mathbf{x}_s, \mathbf{x}_t) &= \frac{P(\mathbf{x}_s, \mathbf{x}_t|\text{same speaker})}{P(\mathbf{x}_s, \mathbf{x}_t|\text{different speakers})} \\ &= \frac{\int p(\mathbf{x}_s, \mathbf{x}_t, \mathbf{z}|\boldsymbol{\omega})\mathrm{d}\mathbf{z}}{\int p(\mathbf{x}_s, \mathbf{z}_s|\boldsymbol{\omega})\mathrm{d}\mathbf{z}_s \int p(\mathbf{x}_t, \mathbf{z}_t|\boldsymbol{\omega})\mathrm{d}\mathbf{z}_t} \\ &= \frac{\int p(\mathbf{x}_s, \mathbf{x}_t|\mathbf{z}, \boldsymbol{\omega})p(\mathbf{z})\mathrm{d}\mathbf{z}}{\int p(\mathbf{x}_s|\mathbf{z}_s, \boldsymbol{\omega})p(\mathbf{z}_s)\mathrm{d}\mathbf{z}_s \int p(\mathbf{x}_t|\mathbf{z}_t, \boldsymbol{\omega})p(\mathbf{z}_t)\mathrm{d}\mathbf{z}_t} \\ &= \frac{\mathcal{N}\left(\left[\mathbf{x}_s^\mathsf{T}\ \mathbf{x}_t^\mathsf{T}\right]^\mathsf{T} \middle| \left[\boldsymbol{\mu}^\mathsf{T}\ \boldsymbol{\mu}^\mathsf{T}\right]^\mathsf{T}, \tilde{\mathbf{W}}\tilde{\mathbf{W}}^\mathsf{T} + \tilde{\boldsymbol{\Sigma}}\right)}{\mathcal{N}\left(\left[\mathbf{x}_s^\mathsf{T}\ \mathbf{x}_t^\mathsf{T}\right]^\mathsf{T} \middle| \left[\boldsymbol{\mu}^\mathsf{T}\ \boldsymbol{\mu}^\mathsf{T}\right]^\mathsf{T}, \text{diag}\{\mathbf{W}\mathbf{W}^\mathsf{T} + \boldsymbol{\Sigma}, \mathbf{W}\mathbf{W}^\mathsf{T} + \boldsymbol{\Sigma}\}\right)} \end{aligned} \tag{2}$$

where $\tilde{\mathbf{W}} = \left[\mathbf{W}^\mathsf{T}\ \mathbf{W}^\mathsf{T}\right]^\mathsf{T}$ and $\tilde{\boldsymbol{\Sigma}} = \text{diag}\{\boldsymbol{\Sigma}, \boldsymbol{\Sigma}\}$. Using Eq. 2 and the standard formula for the inverse of block matrices [25], the log-likelihood ratio score is given by

$$S_{\text{LR}}(\mathbf{x}_s, \mathbf{x}_t) = \text{const} + \mathbf{x}_s^\mathsf{T}\mathbf{Q}\mathbf{x}_s + \mathbf{x}_t^\mathsf{T}\mathbf{Q}\mathbf{x}_t + 2\mathbf{x}_s^\mathsf{T}\mathbf{P}\mathbf{x}_t, \tag{3}$$

where

$$\begin{aligned} &\mathbf{P} = \boldsymbol{\Lambda}^{-1}\boldsymbol{\Gamma}(\boldsymbol{\Lambda} - \boldsymbol{\Gamma}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Gamma})^{-1};\ \ \boldsymbol{\Lambda} = \mathbf{W}\mathbf{W}^\mathsf{T} + \boldsymbol{\Sigma} \\ &\mathbf{Q} = \boldsymbol{\Lambda}^{-1} - (\boldsymbol{\Lambda} - \boldsymbol{\Gamma}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Gamma})^{-1};\ \ \boldsymbol{\Gamma} = \mathbf{W}\mathbf{W}^\mathsf{T}. \end{aligned} \tag{4}$$

2.2. Utterance Partitioning with Acoustic Vector Resampling

The aim of utterance partitioning is to maximize the utilization of target-speaker's information and to increase the influence of speaker-class data on the SVM decision boundary. UP-AVR [12] uses the notion of random resampling in bootstrapping [26] to produce a sufficient number of i-vectors without compromising their representation power. For each conversation, a sequence of acoustic vectors is extracted. Then, the sequence is partitioned into N equal-length segments, and an i-vector is estimated from each segment. If more i-vectors are required, the vector sequence is randomly reshuffled and the partitioning process is repeated to produce another N vectors. If this partitioning-randomization process is repeated R times, $(RN + 1)$ i-vectors can be obtained from a single conversation, where the additional one is obtained from the entire acoustic sequence.

In theory, an infinite number of i-vectors can be obtained when $R \to \infty$. However, when R increases, a segment will contain a significant number of acoustic vectors that also appear in other segments, which results in many similar i-vectors. To avoid this situation, R should be small. In this work, R was limited to 4.

2.3. Empirical Kernels and Empirical Kernel Maps

Eq. 2 shows that PLDA LR scoring uses the information of background speakers implicitly. To make better use of the background information, we derived a speaker-dependent discriminative model for scoring – empirical LR SVM.

Assume that after UP-AVR, H_s i-vectors have been extracted from the enrollment utterance of speaker s. Denote these i-vectors as:

$$\mathcal{X}_s = \{\mathbf{x}_{s,1}, \ldots, \mathbf{x}_{s,H_s}\}. \tag{5}$$

Let's denote the set of background-speaker i-vectors as:[1]

$$\mathcal{X}_b = \{\mathbf{x}_{b,1}, \ldots, \mathbf{x}_{b,B}\}. \quad (6)$$

Then, the SVM score of a test i-vector $\mathbf{x}_t$ is

$$S_{\text{SVM}}(\mathbf{x}_t, \mathcal{X}_s, \mathcal{X}_b) = \sum_{j \in \text{SV}_s} \alpha_{s,j} K(\mathbf{x}_t, \mathbf{x}_{s,j}) - \sum_{j \in \text{SV}_b} \alpha_{s,j} K(\mathbf{x}_t, \mathbf{x}_{b,j}) + d_s \quad (7)$$

where SV_s and SV_b contain the indexes of the support vectors corresponding to the speaker class and impostor class, respectively, and d_s is a speaker-dependent bias.

There are several possibilities for the kernel $K(\cdot,\cdot)$. We focus on the following two cases.

1. ***Empirical LR Kernel I***:

$$K(\mathbf{x}_t, \mathbf{x}_{s,j}) = \mathbb{K}\left(\overrightarrow{S}_{\text{LR}}(\mathbf{x}_t, \mathcal{X}_s), \overrightarrow{S}_{\text{LR}}(\mathbf{x}_{s,j}, \mathcal{X}_s)\right) \quad (8)$$

where

$$\overrightarrow{S}_{\text{LR}}(\mathbf{x}_t, \mathcal{X}_s) = \begin{bmatrix} S_{\text{LR}}(\mathbf{x}_t, \mathbf{x}_{s,1}) \\ S_{\text{LR}}(\mathbf{x}_t, \mathbf{x}_{s,2}) \\ \vdots \\ S_{\text{LR}}(\mathbf{x}_t, \mathbf{x}_{s,H_s}) \end{bmatrix} \quad (9)$$

is an empirical kernel map and $\mathbb{K}(\cdot,\cdot)$ is a standard SVM kernel, e.g., linear or RBF. $\overrightarrow{S}_{\text{LR}}(\mathbf{x}_{s,j}, \mathcal{X}_s)$ can be obtained by replacing $\mathbf{x}_t$ in Eq. 9 with $\mathbf{x}_{s,j}$. Similar formulations apply to $K(\mathbf{x}_t, \mathbf{x}_{b,j})$ in Eq. 7. Note that the empirical feature space is defined by target-speaker's i-vectors through the PLDA model. Because H_s is typically small (17 in this work), the dimension of $\overrightarrow{S}_{\text{LR}}(\mathbf{x}_t, \mathcal{X}_s)$ is low. Therefore, it is possible to use a non-linear kernel for $\mathbb{K}(\cdot,\cdot)$.

2. ***Empirical LR Kernel II***: Let's denote $\mathcal{X} = \{\mathcal{X}_s, \mathcal{X}_b\}$ as the training set for target-speaker s. Then,

$$K(\mathbf{x}_t, \mathbf{x}_{s,j}) = \mathbb{K}\left(\overrightarrow{S}_{\text{LR}}(\mathbf{x}_t, \mathcal{X}), \overrightarrow{S}_{\text{LR}}(\mathbf{x}_{s,j}, \mathcal{X})\right) \quad (10)$$

where

$$\overrightarrow{S}_{\text{LR}}(\mathbf{x}_t, \mathcal{X}) = \begin{bmatrix} S_{\text{LR}}(\mathbf{x}_t, \mathbf{x}_{s,1}) \\ \vdots \\ S_{\text{LR}}(\mathbf{x}_t, \mathbf{x}_{s,H_s}) \\ S_{\text{LR}}(\mathbf{x}_t, \mathbf{x}_{b,1}) \\ \vdots \\ S_{\text{LR}}(\mathbf{x}_t, \mathbf{x}_{b,B'}) \end{bmatrix} \quad (11)$$

where the B' ($B' \leq B$) background i-vectors are selected from the background speaker set $\mathcal{X}_b$. Unlike Empirical LR Kernel I, the score vector in Eq. 11 also contains the LR scores of $\mathbf{x}_t$ with respect to the background i-vectors. As a result, discriminative information between same-speaker pairs $\{\mathbf{x}_t, \mathbf{x}_{s,j}\}_{j=1}^{H_s}$ and different-speaker pairs $\{\mathbf{x}_t, \mathbf{x}_{b,j}\}_{j=1}^{B}$ is embedded in the score vector. Note that the vector size in Eq. 11 is independent of the number of target-speakers. Therefore, the method is scalable to large systems with thousands of speakers.

[1] It is not necessary to apply UP-AVR to background speakers because background i-vectors are abundant.

3. EXPERIMENTS AND RESULTS

3.1. Speech Data and Acoustic Features

The *extended core set* of NIST 2010 Speaker Recognition Evaluation (SRE) was used for performance evaluation. This paper focuses on the interview and microphone speech of the extended core task, i.e., Common Conditions 1, 2, 4, 7 and 9. The interview and microphone speech of male speakers in NIST 2005–2008 SREs were used as development data for training the UBM, total variability matrix, and PLDA model.

An in-house voice activity detector (VAD) [27, 28] was applied to detect the speech regions of each utterance.[2] Briefly, for each conversation side, the VAD uses spectral subtraction with a large over-subtraction factor to remove the background noise. The low energy and high energy regions of the noise-removed speech were used for estimating a decision threshold. This energy-based threshold was then applied to the whole utterance to detect the speech regions.

19 MFCCs together with energy plus their 1st- and 2nd- derivatives were extracted from the speech regions as detected by the VAD, followed by cepstral mean normalization [29] and feature warping [30] with a window size of 3 seconds. A 60-dim acoustic vector was extracted every 10ms, using a Hamming window of 25ms.

3.2. Total Variability Modeling and PLDA

The i-vector systems are based on a gender-dependent UBM with 1024 mixtures. 4,072 microphone utterances from NIST 2005–2008 SREs were used for training the UBM. We selected 9,511 utterances from 191 speakers in NIST 2005–2008 SREs to estimate a total variability matrix with 400 total factors. We used the same data set for training the total variability matrix to estimate the loading matrix of Gaussian PLDA, but excluding those speakers with less than 8 utterances. Similar to [31], we applied within-class covariance normalization (WCCN) [32] and i-vector length normalization before training the PLDA model. The number of latent variables in PLDA was set to 150.

3.3. PLDA LR Scoring versus PLDA SVM Scoring

We considered the classical LR scoring based on Gaussian PLDA model as the baseline (*PLDA* in Table 1 and Fig. 1). For SVM scoring, we selected 633 background speakers (i.e., $B = 633$ in Eq. 6) and used their i-vectors to train an SVM for each target-speaker using the empirical LR kernels described in Section 2.3. Both linear and RBF kernels were used for $\mathbb{K}(\cdot,\cdot)$ in Eqs. 8 and 10. The penalty factor was set to 1.0 for all SVMs.

UP-AVR was applied to both the baseline (PLDA) and SVMs. For the former, we used the RN i-vectors produced by UP-AVR together with the one estimated from the full-length enrollment utterance to represent a target speaker. During verification, given a test utterance, we computed the average PLDA LR score between the i-vector of the test utterance and each of these $(RN+1)$ target-speaker i-vectors. In the experiment, we set $R = 4$ and $N = 4$, resulting in 17 i-vectors per target speaker. For the latter, for each target speaker, 17 i-vectors generated by UP-AVR and 633 background i-vectors were used for training his/her scoring SVM. To reduce scoring time, B' in Eq. 11 was set to 100, which results in score vector dimension of 17 and 117 for Empirical LR Kernel I and II, respectively.

[2] Resources of this VAD, including a Linux program and some segmentation files, can be downloaded from http://bioinfo.eie.polyu.edu.hk/ssvad/.

	Method	Kernel $\mathbb{K}$	EER (%)						MinNDCF					
			CC1	CC2	CC4	CC7	CC9	Mic	CC1	CC2	CC4	CC7	CC9	Mic
1	PLDA	–	1.68	2.75	3.21	9.49	**2.56**	2.89	0.30	0.45	0.44	0.89	0.18	0.47
2	PLDA+UP-AVR	–	1.97	2.97	3.39	10.01	**2.56**	3.15	0.34	0.48	0.49	0.90	0.26	0.51
3	PLDA+UP-AVR+SVM-I	Linear	1.61	2.87	3.18	10.45	3.41	3.00	0.26	0.42	0.41	**0.84**	0.21	0.42
4	PLDA+UP-AVR+SVM-I	RBF	1.61	2.87	2.81	15.61	3.42	3.02	0.26	**0.42**	**0.37**	0.85	**0.16**	**0.42**
5	PLDA+SVM-II	Linear	3.07	5.18	5.22	11.54	4.27	5.16	0.68	0.87	0.81	0.99	0.85	0.86
6	PLDA+SVM-II	RBF	2.78	4.82	4.92	10.60	4.06	4.84	0.45	0.71	0.64	0.92	0.58	0.68
7	PLDA+UP-AVR+SVM-II	Linear	1.66	3.03	3.37	11.73	3.42	3.18	0.40	0.61	0.54	0.84	0.47	0.58
8	PLDA+UP-AVR+SVM-II	RBF	**1.31**	**2.47**	**2.76**	**9.40**	3.42	**2.65**	**0.25**	0.45	0.39	0.87	0.20	0.44

Table 1. Performance of various scoring methods for NIST 2010 SRE (male speakers) under the common conditions that involve microphone recordings. The methods are named by the processes applied to the i-vectors for computing the verification scores. For example, *PLDA+UP-AVR+SVM-I* means that UP-AVR has been applied to create target-speaker i-vectors for training SVMs that use Empirical LR Kernel I (Eq. 8). To highlight the importance of UP-AVR, Rows 5 and 6 show the performance of SVM scoring without UP-AVR. No score normalization was applied in all systems.

Table 1 shows the performance of various scoring methods under the common conditions that involve microphone recordings, and Fig. 1 shows the DET curves achieved by four of the scoring methods. The scoring methods' nomenclature is according to the processes that were applied to the i-vectors. For example, *PLDA+UP-AVR* represents applying UP-AVR to create a number of target-speaker's i-vectors for PLDA LR scoring.

The results suggest that except for CC9, SVM scoring that uses UP-AVR and RBF kernels for $\mathbb{K}$ (Row 8) significantly outperforms PLDA LR scoring (Row 1), demonstrating the advantage of incorporating discriminative information in the empirical feature space. Comparing Rows 3 and 7 reveals that the linear kernel performs better in the empirical LR kernel I (Eq. 8). This result is reasonable because the score vectors $\overrightarrow{S}_{\text{LR}}$ in Eq. 9 only use the target-speaker i-vectors as references; as a result, a linear kernel will suffice. On the other hand, the score vectors in Eq. 11 use both target- and background-speaker i-vectors as references, resulting in more complex score vectors and therefore require non-linear kernels for $\mathbb{K}$.

3.4. UP-AVR for SVM Scoring and LR Scoring

To highlight the importance of UP-AVR in SVM scoring, Rows 5 and 6 of Table 1 show the performance of SVM scoring without UP-AVR, i.e., each SVM was trained by one target-speaker i-vector and 633 background i-vectors. Evidently, without UP-AVR, the performance of SVMs becomes so poor that the error rates are even higher than that of PLDA. A comparison between Rows 6 and 8 further suggests that UP-AVR is indispensable to SVM scoring. UP-AVR not only helps to alleviate the data-imbalance problem in SVM training, but also enriches the information content of the scoring vectors by increasing the number of LR scores derived from the target speaker. However, UP-AVR is not beneficial to LR scoring, as evident by the inferior performance of *PLDA+UP-AVR* in Table 1 and Fig. 1

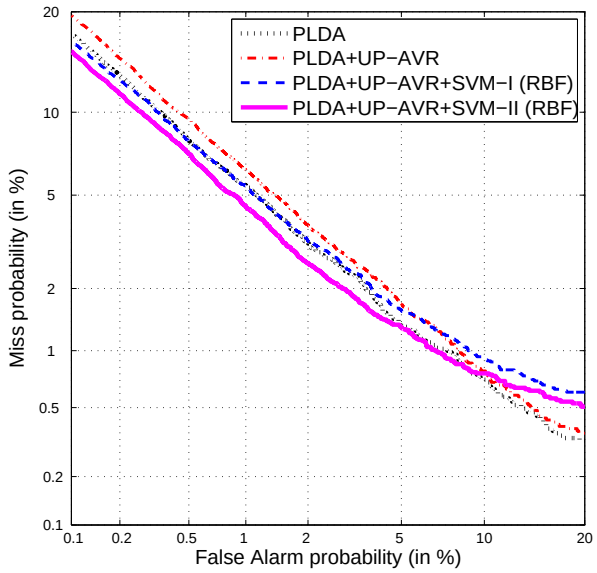

Fig. 1. The DET performance of PLDA LR scoring and SVM scoring using empirical LR kernels under the interview-interview conditions (CC1 and CC2) in NIST 2010 SRE. See Table 1 for the nomenclature of methods in the legend.

4. CONCLUSIONS AND FUTURE WORK

This paper takes the advantage of empirical kernel maps to construct discriminative kernels for SVM scoring under the i-vector based PLDA framework. The paper demonstrates that through empirical kernel maps, the discriminative information of same-speaker and different-speaker i-vector pairs can be captured in both the empirical feature space and the SVM weights. Results show that whenever SVM scoring is applied, utterance partitioning is indispensable because it can help the SVM training algorithm to mitigate the data-imbalance problem.

In NIST 2012 SRE, many of the target speakers have more than one enrollment sessions. As the performance of the empirical kernel maps depends on the number of target-speaker i-vectors and background i-vectors, the large number of enrollment sessions per speaker in NIST 2012 SRE is likely to make SVM scoring outperforms conventional PLDA LR scoring. It is therefore worthwhile to further investigate how the empirical kernel maps and SVM scoring can benefit NIST 2012 SRE.

5. REFERENCES

[1] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, May 2011.

[2] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 15, no. 4, pp. 1435–1447, May 2007.

[3] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A study of inter-speaker variability in speaker verification," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 16, no. 5, pp. 980–988, 2008.

[4] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," in *Proc. of Odyssey: Speaker and Language Recognition Workshop*, Brno, Czech Republic, June 2010.

[5] D. Garcia-Romero and C.Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Interspeech'2011*, 2011, pp. 249–252.

[6] S.J.D. Prince and J.H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 2007, pp. 1–8.

[7] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1–3, pp. 19–41, Jan. 2000.

[8] B. Scholkopf, S. Mika, C. J. C. Burges, P. Knirsch, K. R. Muller, G. Ratsch, and A. J. Smola, "Input space versus feature space in kernel-based methods," *IEEE Trans. on Neural Networks*, vol. 10, no. 5, pp. 1000–1017, Sept. 1999.

[9] H. Xiong, M.N.S Swamy, and M.O. Ahmad, "Optimizing the kernel in the empirical feature space," *IEEE Transactions on Neural Networks*, vol. 16, no. 2, pp. 460–474, 2005.

[10] S. X. Zhang and M. W. Mak, "Optimized Discriminative Kernel for SVM Scoring and Its Application to Speaker Verification," *IEEE Trans. on Neural Networks*, vol. 22, no. 2, pp. 173–185, 2011.

[11] W. Rao and M. W. Mak, "Addressing the data-imbalance problem in kernel-based speaker verification via utterance partitioning and speaker comparison," in *Proc. of Interspeech 2011*, Florence, Aug. 2011, pp. 2717–2720.

[12] M.W. Mak and W. Rao, "Utterance partitioning with acoustic vector resampling for GMM-SVM speaker verification," *Speech Communication*, vol. 53, no. 1, pp. 119–130, Jan. 2011.

[13] W. Rao and M. W. Mak, "Utterance partitioning with acoustic vector resampling for i-vector based speaker verification," in *Odyssey 2012: The Speaker and Language Recognition Workshop*, Singapore, 2012.

[14] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 308–311, May 2006.

[15] W. M. Campbell, D. E. Sturim, D. A. Reynolds, and A. Solomonoff, "SVM based speaker verification using a GMM supervector kernel and NAP variability compensation," in *Proc. ICASSP*, Toulouse, France, May 2006, vol. 1, pp. 97–100.

[16] W.M. Campbell, J.P. Campbell, T.P. Gleason, D.A. Reynolds, and W. Shen, "Speaker verification using support vector machines and high-level features," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 7, pp. 2085–2094, 2007.

[17] W. Rao and M. W. Mak, "Boosting the performance of i-vector based speaker verification via utterance partitioning," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 21, no. 5, pp. 1012–1022, 2013.

[18] S. Cumani, N. Brummer, L. Burget, and P. Laface, "Fast discriminative speaker verification in the i-vector space," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011, pp. 4852–4855.

[19] L. Burget, O. Plchot, S. Cumani, O. Glembek, P. Matejka, and N. Brimmer, "Discriminatively trained probabilistic linear discriminant analysis for speaker verification," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011, pp. 4832–4835.

[20] S. Cumani, O. Glembek, N. Brummer, E. de Villiers, and P. Laface, "Gender independent discriminative speaker recognition in i-vector space," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, 2012, pp. 4361–4364.

[21] D. Sturim, D. A. Reynolds, E. Singer, and J. P. Campbell, "Speaker indexing in large audio databases using anchor models," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, 2001, vol. 1, pp. 429–432.

[22] M. Collet, D. Charlet, and F. Bimbot, "Speaker tracking by anchor models using speaker segment cluster information," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, 2006, pp. 1009–1012.

[23] E. Noor and H. Aronowitz, "Efficient language identification using anchor models and support vector machines," in *Odyssey 2006: The Speaker and Language Recognition Workshop*, 2006, pp. 1–6.

[24] C.M. Bishop, *Pattern recognition and machine learning*, springer, New York, 2006.

[25] K. B. Petersen and M. S. Pedersen, "The matrix cookbook," Oct 2008.

[26] B. Efron and G. Gong, "A leisurely look at bootstrap, the jackknife, and cross-validation," *The American Statistician*, vol. 37, no. 1, pp. 36–48, 1983.

[27] M. W. Mak and H. B. Yu, "Robust voice activity detection for interview speech in nist speaker recognition evaluation," in *Proc. APSIPA ASC 2010*, Singapore, 2010.

[28] H.B. Yu and M.W. Mak, "Comparison of voice activity detectors for interview speech in nist speaker recognition evaluation," in *Interspeech*, 2011, pp. 2353–2356.

[29] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," *J. Acoust. Soc. Am.*, vol. 55, no. 6, pp. 1304–1312, Jun. 1974.

[30] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proc. Odyssey: The Speaker and Language Recognition Workshop*, Crete, Greece, Jun. 2001, pp. 213–218.

[31] M. McLaren, M.I. Mandasari, and D.A. Leeuwen, "Source normalization for language-independent speaker recognition using i-vectors," in *Odyssey 2012: The Speaker and Language Recognition Workshop*, 2012, pp. 55–61.

[32] A. Hatch, S. Kajarekar, and A. Stolcke, "Within-class covariance normalization for SVM-based speaker recognition," in *Proc. of the 9th International Conference on Spoken Language Processing*, Pittsburgh, PA, USA, Sep. 2006, pp. 1471–1474.