On Complementarity of State-of-the-art Speaker Recognition Systems

Lukáš Machlica, Zbyněk Zajíc, Luděk Müller
Department of Cybernetics, Faculty of Applied Sciences, University of West Bohemia
Pilsen, Czech Republic
machlica@kky.zcu.cz, zzajic@kky.zcu.cz, muller@kky.zcu.cz

***Abstract*— *In this paper recent methods used in the task of Speaker Recognition (SR) are reviewed and their complementarity is analysed. At first, methods based on Supervectors (SVs) related to Gaussian Mixture Models (GMMs) and Support Vector Machines (SVMs) used as a discriminative model are described along with the Nuisance Attribute Projection (NAP). NAP was proposed to suppress undesirable influences of high channel variabilities between several sessions of a speaker. Next, recent methods focusing on the extraction of so called i-vectors (low dimensional representations of GMM based SVs) are discussed. The space in which i-vectors lie is denoted the Total Variability Space (TVS) since it contains both between-speaker and session/channel variabilities. Once i-vectors have been extracted a Probabilistic Linear Discriminant Analysis (PLDA) model is trained in the TVS. In the training phase of PLDA the TVS is decomposed to a channel and a speaker subspace, hence each i-vector is supposed to be composed from a speaker identity component and a channel component. The complementarity of PLDA and SVM based modelling techniques is examined utilizing the linear logistic regression as a fusion tool used to combine the verification scores of individual systems leading to significant reductions in error rates of the SR system. The results are presented on the NIST SRE 2008 and NIST SRE 2010 corpora.*

Keywords*— *SVM, NAP, i-vector, PLDA, fusion.

I. Introduction

Gaussian Mixture Models (GMMs) introduced in [1] dominated the task of Speaker Recognition (SR) for more than a decade. Nowadays, GMMs play still an important role in the state-of-the-art speaker recognition systems, however they are used mainly to delimit and split up the feature space according to a level of significance, and to extract data statistics related to distinct parts of the feature space. This is done via estimation of an Universal Background Model (UBM) comprising many GMM components and trained on a huge amount of development data. All the acoustic conditions in which the system will be used should be covered.

Subsequently, given an UBM, statistics of extracted feature vectors of a speaker related to distinct parts of the feature space are estimated, and a Supervector (SV) formed by concatenation of statistics from different parts of the space (related to individual Gaussians in the UBM) is formed yielding a SV of substantially high dimension. Methods of SV extraction will be discussed in Section III.

Simultaneously two techniques to handle the high dimensional SVs were proposed. The first was based on Support Vector Machine (SVM) as a discriminative trainer [2], which has good generalization properties and is well suited for the task of modelling when only a few (in the case of SVs often only one) example/vectors of a class/speaker are available. Since both generative (UBM/GMM) and discriminative (SVM) modelling are utilized the techniques comprising GMM based SVs and SVMs are also known as *hybrid modelling*. The concept of SVs and SVM was further extended by the Nuisance Attribute Projection (NAP) [3], which is used to suppress undesirable channel variabilities between sessions of one speaker and will be described in Section IV.

The second technique was based on Factor Analysis (FA) and generative modelling. The idea was that since the dimensionality of SVs is in comparison with the number of development speakers very high, many dimensions have to be correlated with each other. Hence, the true information has to lie in a much lower subspace. Moreover, since several sessions of one speaker are available one could determine not only the speaker identity subspace, but also the channel/session subspace, which should be also of a lower dimension. These principles were incorporated into a method called Joint Factor Analysis (JFA) [4]. However, experiments in [5] have shown, that the channel/session subspace does still contain some substantial information concerning the identity of a speaker. Therefore, JFA was extended to the concept of i-vectors [6], where both subspaces are merged (they are no longer distinguished in the model) forming a Total Variability Space (TVS), see Section VI.

Independently on JFA a method called Probabilistic Linear Discriminant Analysis (PLDA) has been developed in the computer vision to tackle the problem of face recognition [7]. PLDA does a similar job as JFA, it decomposes the feature space to a speaker and channel dependent subspaces, but rather than GMM based SVs ordinary feature vectors are utilized (to understand the difference see Section VI and Section VII, and note the use of matrix $\boldsymbol{N}_s$ which does the weighting of distinct dimensional blocks). Hence, PLDA is well suited as a verification tool for an i-vector based system and will be briefly discussed in Section VII.

The goal of this paper is to examine the complementarity of methods based on SVM modeling, i-vector extraction and PLDA modeling. For this purpose a linear fusion tool will be utilized as described in Secction VIII. The results found in Section VIII-C are presented utilizing recent Speaker Recognition Evaluations (SREs) conducted by NIST.

II. Universal Background Model (UBM)

UBM used in the task of SR is a Gaussian Mixture Model (GMM) trained on a huge amount of development data. It should reflect the acoustic conditions of the environment, in which the speaker recognition system is used. UBM consists of a set of parameters $\boldsymbol{\lambda} = \{\omega_m, \boldsymbol{\mu}_m, \boldsymbol{C}_m\}_{m=1}^{M}$, where M is the number of Gaussians in the UBM, ω_m, $\boldsymbol{\mu}_m$, $\boldsymbol{C}_m$ are the weight, mean and covariance of the m^{th} Gaussian, respectively. The most important statistic related to the m^{th} Gaussian of the UBM and a set of T_s feature vectors $\boldsymbol{O}_s = \{\boldsymbol{o}_{st}\}_{t=1}^{T_s}$ related to the s^{th} speaker is

$$\gamma_m(\boldsymbol{o}_{st}) = \frac{\omega_m \mathcal{N}(\boldsymbol{o}_{st}; \boldsymbol{\mu}_m, \boldsymbol{C}_m)}{\sum_{m=1}^{M} \omega_m \mathcal{N}(\boldsymbol{o}_{st}; \boldsymbol{\mu}_m, \boldsymbol{C}_m)}, \tag{1}$$

where $\mathcal{N}(\boldsymbol{o}_{st}; \boldsymbol{\mu}_m, \boldsymbol{C}_m)$ is the Gaussian probability density function with mean $\boldsymbol{\mu}_m$ and covariance $\boldsymbol{C}_m$. For further use let $D = \dim(\boldsymbol{o}_{st})$ be the dimension of feature vectors.

III. Supervectors (SVs)

The term supervector (SV) used in SR is related to a high dimensional vector obtained by the concatenation of several vectors. Once the UBM was trained, supervectors

$$\begin{aligned} \boldsymbol{b}_s &= \sum_{t=1}^{T_s} \left[\gamma_1(\boldsymbol{o}_{st})\boldsymbol{o}_{st}^{\mathrm{T}}, \ldots, \gamma_M(\boldsymbol{o}_{st})\boldsymbol{o}_{st}^{\mathrm{T}}\right]^{\mathrm{T}}, \\ \boldsymbol{n}_s &= \sum_{t=1}^{T_s} \left(\left[\gamma_1(\boldsymbol{o}_{st}), \ldots, \gamma_M(\boldsymbol{o}_{st})\right]^{\mathrm{T}} \otimes \boldsymbol{1}_D \right), \end{aligned} \tag{2}$$

can be extracted, both are of size $DM \times 1$, $\otimes$ is the Kronecker product, $\boldsymbol{1}_D$ is a D dimensional vector of ones, and let $\boldsymbol{m}_0 = [\boldsymbol{\mu}_1^{\mathrm{T}}, \boldsymbol{\mu}_2^{\mathrm{T}}, \ldots, \boldsymbol{\mu}_M^{\mathrm{T}}]^{\mathrm{T}}$ be the SV constructed by the concatenation of the UBM means.

A. GMM-mean Supervector (GSV)

GSV was proposed in [2], and it is composed of means of an Maximum A-Posteriory (MAP) adapted UBM. GSV (and also the MAP adaptation of UBM means) can be expressed as

$$\boldsymbol{\psi}^s_{\text{GSV}} = \tau \boldsymbol{m}_s + (1-\tau)\boldsymbol{m}_0, \tag{3}$$

$$\boldsymbol{m}_s = \boldsymbol{N}_s^{-1}\boldsymbol{b}_s, \tag{4}$$

where $\boldsymbol{m}_s$ is the new Maximum Likelihood (ML) estimate of $\boldsymbol{m}_0$ given the dataset $\boldsymbol{O}_s$, $\boldsymbol{N}_s$ is a diagonal matrix with $\boldsymbol{n}_s$ on its diagonal, and τ is an empirically set parameter controlling the balance between UBM parameters $\boldsymbol{m}_0$ and the new ML estimate $\boldsymbol{m}_s$. Note that for each speaker only one SV is extracted no matter how many feature vectors are available.

B. Generalized Linear Discriminant Sequence (GLDS)

GLDS was proposed in [8]. It is based on a vector function that transforms directly the feature vectors (UBM is not involved). The SV has the form

$$\boldsymbol{\psi}^s_{\text{GLDS}} = \frac{1}{T}\sum_{t=1}^{T_s} \boldsymbol{\varphi}(\boldsymbol{o}_{st}; k)\,, \tag{5}$$

where $\boldsymbol{\varphi}(\boldsymbol{o}_{st}; k)$ represents a monomial expansion of a feature vector $\boldsymbol{o}_{st}$ up to the k^{th} order, e.g. for a monomial expansion of a D dimensional feature vector $\boldsymbol{o} = [o_1, o_2, \ldots, o_D]^{\mathrm{T}}$ up to the second order we get

$$\boldsymbol{\varphi}(\boldsymbol{o}; k=2) = [1, o_1, \ldots, o_D, o_1^2, o_1o_2, \ldots, o_1o_D, \tag{6}$$

$$o_2^2, o_2o_3, \ldots, o_2o_D, o_3^2, \ldots, o_D^2], \tag{7}$$

where $\dim(\boldsymbol{\psi}_{\text{GLDS}}) = ((D+k)!)(\,D!\,k!\,)$. After substituting (7) into (5) one can notice, that the mapping (5) comprises first- and second-order moments – the means and correlations of feature vectors [9]. Again data statistics are collected (as in the case of GSV), however now also statistics of higher order may be acquired. E.g if only monomials up to order two are required, the GLDS mapping is build from the mean and concatenated rows of covariance matrix acquired assuming a single component UBM/GMM. Note again that only one SV is extracted for each speaker.

IV. Nuisance Attribute Projection (NAP)

In cases when several recordings of a speaker are available, recorded on distinct channels, the channel/session information can be utilized in order to suppress high within-speaker deviations [3].

The objective function minimized in NAP is given as

$$J_{\text{NAP}}(\boldsymbol{P}) = \sum_{i=1}^{N-1}\sum_{j=i+1}^{N} w_{ij}\|\boldsymbol{P}(\boldsymbol{x}_i - \boldsymbol{x}_j)\|^2, \tag{8}$$

where $\boldsymbol{x}_i$ is a SV of dimension D_x, N is the number of SVs in the development set, $w_{ij} = 1$ if both $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ come from the same speaker, and 0 otherwise. $\boldsymbol{P} = \boldsymbol{I} - \boldsymbol{F}_\perp\boldsymbol{F}_\perp^{\mathrm{T}}$ is a projection matrix, $\boldsymbol{F}_\perp$ is a $D_x \times D_c$ matrix of low rank D_c, where $D_c \ll D_x$, columns of $\boldsymbol{F}_\perp$ are orthonormal, thus $\boldsymbol{F}_\perp^{\mathrm{T}}\boldsymbol{F}_\perp = \boldsymbol{I}$ and they span the subspace that is going to be projected out. It is easy to see that the properties of $\boldsymbol{P}$ are: $\boldsymbol{P}^2 = \boldsymbol{P}$ ($\boldsymbol{P}$ is idempotent) and $\boldsymbol{P} = \boldsymbol{P}^{\mathrm{T}}$ ($\boldsymbol{P}$ is symmetric). It can be shown [10] that the objective function (8) can be expressed as

$$J_{\text{NAP}}(\boldsymbol{P}) = \text{tr}(\boldsymbol{P}\boldsymbol{C}_{\text{W}}) = \text{tr}(\boldsymbol{C}_{\text{W}}) - \text{tr}(\boldsymbol{F}_\perp^{\mathrm{T}}\boldsymbol{C}_{\text{W}}\boldsymbol{F}_\perp), \tag{9}$$

$$\boldsymbol{C}_{\text{W}} = \sum_{s=1}^{S} H_s \sum_{h=1}^{H_s} (\boldsymbol{x}_{sh} - \bar{\boldsymbol{x}}_s)(\boldsymbol{x}_{sh} - \bar{\boldsymbol{x}}_s)^{\mathrm{T}}, \tag{10}$$

$$\bar{\boldsymbol{x}}_s = \sum_{h=1}^{H_s} \boldsymbol{x}_{sh}, \tag{11}$$

where H_s is the number of session of speaker s, S is the number of speakers in the development set, and $\boldsymbol{C}_{\text{W}}$ is the weighted within-speaker covariance computed on the development set of speakers. The objective (9) is minimized when columns of $\boldsymbol{F}_\perp$ are formed by eigenvectors of $\boldsymbol{C}_{\text{W}}$ corresponding to the D_c largest eigenvalues (highest variance is projected out).

V. Support Vector Machine (SVM)

SVM is a binary classifier, where the decision boundary between two classes is given by a linear hyperplane and the task is to find a separating hyperplane so that the margin between the classes is maximized [11]. Whenever a decision of a classification depends only on a dot product of two vectors, the dot product can be replaced by a scalar *kernel function* $K(\boldsymbol{x}_1, \boldsymbol{x}_2)$, which has to satisfy certain restrictions called Mercer's conditions. These conditions specify requirements under which the output of the kernel function can be thought of as an output of a dot product of two vectors. Thus $K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \phi(\boldsymbol{x}_1)^{\mathrm{T}}\phi(\boldsymbol{x}_2)$, where $\phi(\boldsymbol{x}_i)$ is a vector function that maps $\boldsymbol{x}_i$ to some high dimensional vector (even of infinite dimension). The SVM decision function can be written as

$$f(\boldsymbol{x}_i) = \sum_{n=1}^{L} \alpha_n y_n K(\boldsymbol{x}_n, \boldsymbol{x}_i) + q, \tag{12}$$

and if the kernel function is linear $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j$ we get

$$f(\boldsymbol{x}_i) = \left(\sum_{n=1}^{L} \alpha_n y_n \boldsymbol{x}_n^{\mathrm{T}}\right)\boldsymbol{x}_i + q = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + q, \tag{13}$$

where L is the number of support vectors, which combination (not necessary linear) forms the boundary, q is an offset, $\alpha_n > 0$, L and q are learned during the training process of SVM, $y_n \in \{-1, 1\}$ are the class labels, and $\boldsymbol{x}_i$ is the vector which class pertinence has to be determined, e.g. $y_i = \operatorname{sign} f(\boldsymbol{x}_i)$. SVM is trained iteratively utilizing some optimization algorithm [12]. Note that if kernel function is linear only the normal vector $\boldsymbol{w}$ and offset q of the decision boundary have to be stored, but if this is not the case all the support vectors have to be stored and in the decision process the kernel function has to be evaluated L times for each new vector in question.

VI. Extraction of i-vectors

The concept of i-vectors is closely related to a very effective technique called Joint Factor Analysis (JFA) introduced in [4]. Both JFA and i-vectors work with supervectors from (2), hence they are related to a UBM. JFA tries to find (preferably distinct) subspaces responsible for most of the session and speaker variabilities, whereas in the concept of i-vectors these variabilities are not distinguished, only an assumption is met that they can be explained in an sufficient amount by variations of low dimensional hidden variables called identity vectors (i-vectors) [5].

The (generative) model has the form

$$\boldsymbol{\psi}_s = \boldsymbol{m}_0 + \boldsymbol{T}\boldsymbol{w}_s + \boldsymbol{\epsilon}, \tag{14}$$

$$\boldsymbol{w}_s \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \;\; \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}) \tag{15}$$

where $\boldsymbol{w}_s$ is the D_w dimensional i-vector following standard normal distribution extracted from feature vectors of the s^{th} speaker, $\boldsymbol{T}$ is the *total variability space matrix* of size $DM \times D_w$, $\boldsymbol{m}_0$ is the mean vector of $\boldsymbol{\psi}_s$ (often mean supervector of UBM is taken instead as a good approximation), and $\boldsymbol{\epsilon}$ is a random variable describing the residual noise following normal distribution with zero mean and diagonal covariance $\boldsymbol{\Sigma}$ (its diagonal blocks are often composed from the covariances $\boldsymbol{C}_1, \ldots, \boldsymbol{C}_m$ of the UBM). Note that the matrix $\boldsymbol{T}$ should encompass both the between-speaker and the within-speaker (i.e. channel/session) variabilities.

A. Training

In order to train the i-vector extractor at first supervectors (2) are extracted for each speaker and each session of a speaker. A crucial assumption is made that each session of a speaker is in fact another speaker, hence within- and between-covariance of a speaker is not distinguished. Now, two steps are iterated in a sequence until predetermined number of iterations is reached:

1) for each s use previous estimate of $\boldsymbol{T}$ to extract new i-vector

$$\boldsymbol{w}_s = (\boldsymbol{I} + \boldsymbol{T}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{N}_s\boldsymbol{T})^{-1}\boldsymbol{T}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\bar{\boldsymbol{b}}_s, \tag{16}$$

2) let $\boldsymbol{Z} = \left(\boldsymbol{I} + \boldsymbol{T}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{N}_s\boldsymbol{T}\right)^{-1}$; use newly extracted i-vectors to compute block-wise a new estimate of $\boldsymbol{T}$

$$\boldsymbol{T}_m = \left(\sum_{s=1}^{S} \bar{\boldsymbol{b}}_{sm}\boldsymbol{w}_s^{\mathrm{T}}\right)\left(\sum_{s=1}^{S} \boldsymbol{N}_{sm}\left(\boldsymbol{w}_s\boldsymbol{w}_s^{\mathrm{T}} + \boldsymbol{Z}\right)\right)^{-1}, \tag{17}$$

where $\boldsymbol{N}_s$ is a diagonal matrix with $\boldsymbol{n}_s$ on its diagonal, $\bar{\boldsymbol{b}}_s = \boldsymbol{b}_s - \boldsymbol{N}_s\boldsymbol{m}_0$ is the centred version of $\boldsymbol{b}_s$ around the mean $\boldsymbol{m}_0$, and the index m in $\boldsymbol{T}_m$, $\bar{\boldsymbol{b}}_{sm}$, $\boldsymbol{n}_{sm}$ (and $\boldsymbol{N}_{sm}$) refers to blocks of $\boldsymbol{T}$, $\bar{\boldsymbol{b}}_s$, $\boldsymbol{n}_s$ (and thus to $\boldsymbol{N}_{sm}$) of sizes $D \times D_w$, $D \times 1$, $D \times 1$, respectively. Hence, $\boldsymbol{T}^{\mathrm{T}} = [\boldsymbol{T}_1^{\mathrm{T}}, \boldsymbol{T}_2^{\mathrm{T}}, \ldots, \boldsymbol{T}_{sM}^{\mathrm{T}}]$, $\bar{\boldsymbol{b}}_s^{\mathrm{T}} = [\bar{\boldsymbol{b}}_{s1}^{\mathrm{T}}, \bar{\boldsymbol{b}}_{s2}^{\mathrm{T}}, \ldots, \bar{\boldsymbol{b}}_{sM}^{\mathrm{T}}]$ and $\boldsymbol{n}_s^{\mathrm{T}} = [\boldsymbol{n}_{s1}^{\mathrm{T}}, \boldsymbol{n}_{s2}^{\mathrm{T}}, \ldots, \boldsymbol{n}_{sM}^{\mathrm{T}}]$. One can update also $\boldsymbol{\Sigma}$, for details see [13]. Note that for each session of a speaker one i-vector is extracted. Moreover, the training procedure is in fact the same as for parameters of a model of Factor Analysis (FA) differing only in the presence of $\boldsymbol{N}_s$ in estimation formulas (16), (17). If $\boldsymbol{N}_s$ would equal the identity matrix $\boldsymbol{I}$ the training procedure would be identical to the estimation procedure of parameters of a FA model, which form is identical to (14).

VII. Probabilistic Linear Discriminant Analysis (PLDA)

PLDA was introduced in [7] for the task of face recognition in the image processing. However, it is well suited also as a verification tool working with i-vectors presented in the previous section. The model is similar to JFA, it incorporates the idea of within- and between-speaker spaces, but instead of working with high dimensional supervectors (2) it can handle the low dimensional i-vectors $\boldsymbol{w}_s$. Thus, while in the i-vector extraction phase no distinction was made between the speaker and the session space, PLDA incorporates such a knowledge and constructs a new generative model in the total variability space.

The generative PLDA model can be expressed as

$$\boldsymbol{w}_{sh} = \boldsymbol{m}_w + \boldsymbol{F}\boldsymbol{z}_s + \boldsymbol{G}\boldsymbol{r}_{sh} + \boldsymbol{\epsilon}, \tag{18}$$

$$\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{S}), \tag{19}$$

$$\boldsymbol{z}_s, \boldsymbol{r}_{sh} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \tag{20}$$

where $\boldsymbol{m}_w$ is the mean of $\boldsymbol{w}_{sh}$, columns of $\boldsymbol{F}$ span the between-speaker space (speaker identity space), $\boldsymbol{z}_s$ of dimension D_z are coordinates in this space and they do not change across sessions of one speaker, columns of $\boldsymbol{G}$ span the channel space, $\boldsymbol{r}_{sh}$ of dimension D_r are the session dependent speaker factors, and $\boldsymbol{\epsilon}$ is a residual noise factor following normal distribution with zero mean and diagonal covariance $\boldsymbol{S}$. Both latent variables $\boldsymbol{z}_s$, $\boldsymbol{r}_{sh}$ follow standard normal distribution and they are assumed to be independent. It is a common and reasonable assumption that $D_z, D_r < D_w$ and $D_z + D_r \approx D_w$.

A. PLDA Training

In [7] following estimation algorithm of model parameters $\boldsymbol{F}$, $\boldsymbol{G}$ and $\boldsymbol{S}$ was proposed. Let

$$\hat{\boldsymbol{w}}_s = [\boldsymbol{w}_{s1} - \boldsymbol{m}_w, \boldsymbol{w}_{s2} - \boldsymbol{m}_w, \ldots, \boldsymbol{w}_{sH_s} - \boldsymbol{m}_w]^{\mathrm{T}}, \tag{21}$$

$$\hat{\boldsymbol{z}}_s = [\boldsymbol{z}_s, \boldsymbol{r}_{s1}, \boldsymbol{r}_{s2}, \ldots, \boldsymbol{r}_{sH_s}]^{\mathrm{T}}, \tag{22}$$

$$\boldsymbol{A}_{H_s} = \begin{bmatrix} \boldsymbol{F} & \boldsymbol{G} & \boldsymbol{0} & \ldots & \boldsymbol{0} \\ \boldsymbol{F} & \boldsymbol{0} & \boldsymbol{G} & \ldots & \boldsymbol{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{F} & \boldsymbol{0} & \boldsymbol{0} & \ldots & \boldsymbol{G} \end{bmatrix}, \tag{23}$$

$$\hat{\boldsymbol{\epsilon}}_{H_s} = [\boldsymbol{\epsilon}_1, \boldsymbol{\epsilon}_2, \ldots, \boldsymbol{\epsilon}_{H_s}]^{\mathrm{T}}, \tag{24}$$

$$\hat{\boldsymbol{\Sigma}}_{H_s} = \begin{bmatrix} \boldsymbol{S} & \boldsymbol{0} & \ldots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{S} & \ldots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \ldots & \boldsymbol{S} \end{bmatrix}, \tag{25}$$

where $\hat{\boldsymbol{\epsilon}} \sim \mathcal{N}(\boldsymbol{0}, \hat{\boldsymbol{\Sigma}}_{H_s})$, yielding a system of equations for each speaker, which can be written in a compact form as

$$\hat{\boldsymbol{w}}_s = \boldsymbol{A}_{H_s}\hat{\boldsymbol{z}}_s + \hat{\boldsymbol{\epsilon}}_{H_s}. \tag{26}$$

Note that $\boldsymbol{A}_{H_s}$ and $\hat{\boldsymbol{\Sigma}}_{H_s}$ depend on the number of sessions H_s of a speaker, therefore their size changes whenever H_s changes. Equation (26) is a standard Factor Analysis (FA) problem, the estimation process is almost identical to the estimation process described in Section VI-A with $\boldsymbol{N}_s = \boldsymbol{I}$, however some additional decompositions of matrices have to be carried out in order to get $\boldsymbol{F}$, $\boldsymbol{G}$ and $\boldsymbol{S}$ instead of $\boldsymbol{A}_{H_s}$ and $\hat{\boldsymbol{\Sigma}}_{H_s}$. The full description of the estimation algorithm is out of the scope of this paper, for details see [7].

B. Verification

Once the PLDA model parameters $\boldsymbol{F}$, $\boldsymbol{G}$ and $\boldsymbol{S}$ were estimated the task is to assign a verification score to given two i-vectors $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$. For this purpose two hypotheses are tested, namely

- hypotheses $\mathcal{H}_s$: $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ share the same identity
- hypotheses $\mathcal{H}_d$: the identity of $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ differs

The log-likelihood ratio is given as

$$\begin{aligned} \mathrm{LLR}(\boldsymbol{w}_1, \boldsymbol{w}_2) &= \log \frac{p(\boldsymbol{w}_1, \boldsymbol{w}_2|\mathcal{H}_s)}{p(\boldsymbol{w}_1|\mathcal{H}_d)p(\boldsymbol{w}_2|\mathcal{H}_d)} = \\ &= \log \mathcal{N}\left(\begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{bmatrix}; \begin{bmatrix} \boldsymbol{m}_w \\ \boldsymbol{m}_w \end{bmatrix}, \begin{bmatrix} \boldsymbol{C}_w & \boldsymbol{C}_F \\ \boldsymbol{C}_F & \boldsymbol{C}_w \end{bmatrix} \right) \\ &- \log \mathcal{N}\left(\begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{bmatrix}; \begin{bmatrix} \boldsymbol{m}_w \\ \boldsymbol{m}_w \end{bmatrix}, \begin{bmatrix} \boldsymbol{C}_w & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{C}_w \end{bmatrix} \right), \end{aligned} \tag{27}$$

where $\boldsymbol{C}_w = \boldsymbol{F}\boldsymbol{F}^{\mathrm{T}} + \boldsymbol{G}\boldsymbol{G}^{\mathrm{T}} + \boldsymbol{S}$ and $\boldsymbol{C}_F = \boldsymbol{F}\boldsymbol{F}^{\mathrm{T}}$. Note that in this verification scenario we do not care about the form of the decomposition of $\boldsymbol{w}_1$ or $\boldsymbol{w}_2$ (latent variables $\boldsymbol{z}_s$, $\boldsymbol{r}_{sh}$ stay unknown), the question stated is whether the two vectors share the same identity given the subspaces generated by $\boldsymbol{F}$ and $\boldsymbol{G}$. Hence two vectors can be compared in a very simple and effective way.

VIII. Experiments

In order to investigate the complementarity of SVM based systems and i-vector based system, outputs (verification scores) of these systems will be fused. For this purpose the Logistic Linear Regression (LLR) from the FoCal tool kit [14] will be utilized. Hence, the fused score $score_{\mathrm{F}}(\boldsymbol{O}_s, \boldsymbol{O}_q)$ will be given as a linear combination of scores obtained from individual systems:

$$\begin{aligned} score_{\mathrm{F}}(\boldsymbol{O}_s, \boldsymbol{O}_q) = \xi_1 f_{\mathrm{GSV}}(\boldsymbol{O}_s, \boldsymbol{O}_q) + \xi_2 f_{\mathrm{GLDS}}(\boldsymbol{O}_s, \boldsymbol{O}_q) + \\ + \xi_3 f_{\mathrm{iVEC+PLDA}}(\boldsymbol{O}_s, \boldsymbol{O}_q) + \xi_4, \end{aligned} \tag{28}$$

where $f_{\mathrm{XY}}(\boldsymbol{O}_s, \boldsymbol{O}_q)$ is the output (verification score) of system XY given a set of feature vectors of each speakers s and q, and $\boldsymbol{\xi} = [\xi_1, \ldots, \xi_4]$ is the vector of fusion coefficients. Let us summarize the main ideas and dissimilarities of methods described in this paper:

1) i-vectors and PLDA model try to find a low dimensional representation of a Supervector (SV) similar to GSV, moreover they try to decompose the feature space into speaker- and session-dependent parts
2) i-vectors and PLDA model are generative and do not discriminate between speakers, whereas SVM as a discriminative classifier does; note that even if PLDA is a discriminative model it discriminates between the speaker- and the channel-subspace
3) presented SVs incorporate different kind of information; in the case of GSV vectors pointing to positions in the feature space with increased concentration of feature vectors are used; in the case of GLDS the covariance

and higher order moments of the whole speaker's data set are extracted

Therefore the complementarity of presented methods should be preserved. The performance and complementarity of the speaker recognition systems will be tested on the male telephone conversation speech taken from Speaker Recognition Evaluations (SREs) conducted by NIST, more precisely on NIST SRE 2008 (NIST08) and NIST SRE 2010 (NIST10). To train the Fusion Coefficients (FCs) data from NIST08 will be utilized, and the learned FCs will be then used to fuse outputs of systems trained for NIST10.

In NIST08 648 target speakers and 1535 test speakers were present yielding 16 968 trials in total (short2-short3 trials[1]) to be scored, and in the case of NIST10 1394 target speakers and 2474 test speakers were given yielding 74 762 trials in total (core-core trials[2]). The duration of all the test and target recordings in both corpora was approximately 5 minutes including the silence.

A. Feature Extraction

The feature extraction was based on Linear Frequency Cepstral Coefficients (LFCCs), Hamming window of length 25 ms was used, the shift of the window was set to 10 ms, 25 triangular filter banks were spread linearly across the frequency spectrum, and 20 LFCCs were extracted, delta coefficients were added leading to a 40 dimensional feature vector. Also the Feature Warping (FW) normalization procedure was applied utilizing a sliding window of length 3 seconds. Just before FW voice activity detection, based on detection of energies in filter banks located in the frequency domain, was carried out in order to discard non-speech frames. All the feature vectors were at the end down-sampled by a factor of 2.

B. System Set-up

Development corpora NIST SRE 2004 (NIST04), NIST SRE 2005 (NIST05), NIST SRE 2006 (NIST06), Switchboard 1 Release 2 (SW1), Switchboard 2 Phase 3 (SW2), Switchboard Cellular Audio Part 1 and Part 2 (SWC) and Fisher English Training Speech Part 1 and Part 2 (FSH) were used. The overall number of male speakers in NIST04, NIST05, NIST06 was 465 with approximately 8 session for each speaker, in SW1, SW2, SWC 659 male speaker with approx. 11 sessions were present, and in FSH the number of speakers was 1612 with at most 3 sessions each. The data source of all the recordings was telephone conversation and the duration of each recording including the silence was approx. 5 minute, but in FSH the length of recordings varied from 6 to 12 minutes. The summary is given in Table 1.

The number of Gaussians in the UBM was set to 1024. The size of the Total Variability Space (TVS) matrix $\boldsymbol{T}$ was set to $1024 * 40 \times 800$, hence the latent dimension (dimension of i-vectors) was $D_w = 800$. UBM was trained on all the development corpora and so was the TVS matrix. The dimension of the between-speaker subspace in the PLDA model was set to $D_z = 500$ and the dimension of the session/channel space was set to $D_r = 500$, thus both $\boldsymbol{F}$ and $\boldsymbol{G}$ were of size 800×500. In the training phase of the PLDA model the corpus FSH was left out since maximum number of sessions per speaker was 3.

The dimension of supervector was dim $(\boldsymbol{\psi}_{\text{GSV}}) = 1024 * 40 = 40960$, and since in the case of GLDS monomial expansion up to the order 3 was used dim $(\boldsymbol{\psi}_{\text{GLDS}}) = ((1024+3)!)(\,1024!\,3!\,) = 12341$. NAP matrix was trained on SW1, SW2 and SWC since most session were available for this corpora. The rank of $\boldsymbol{F}_\perp$ in the case of $\boldsymbol{\psi}_{\text{GSV}}$ was $D_c = 256$, and since the dimension of $\boldsymbol{\psi}_{\text{GLDS}}$ is lower in the case of GLDS $D_c = 64$. SVMTorch [12] was utilized to train the SVM models, and only simple linear kernels $K = \boldsymbol{\psi}_{\text{GSV}}^{\text{T}} \boldsymbol{\psi}_{\text{GSV}}$, $K = \boldsymbol{\psi}_{\text{GLDS}}^{\text{T}} \boldsymbol{\psi}_{\text{GLDS}}$ were used. Since SVM is a binary classifier we used the one-against-all approach, hence all speakers from NIST04, NIST05, NIST06 were taken as negative examples when training a SVM model of each speaker. The SVM verification score was given as the output of (13). Note that NAP was applied only to the population of background SVs since linear kernel is in use and the NAP projection matrix has the idempotent property $P^2 = P$ (for more details see Section IV).

C. Results

Results are given in Table 2 in terms of Equal Error Rate (EER) and also the minimum of the Decision Cost Function (minDCF) is reported. In order to compute the value of

Table 1. Summary of number of recordings, average number of sessions and number of speakers in distinct corpora.

corpus ID	recordings	male sessions	speakers
NIST04,05,06	3787	8	465
SW1	2342	11	211
SW2	2183	10	216
SWC	2707	12	232
FSH	4923	3	1612
overall	15942	-	2736

Table 2. Equal Error Rates (EERs) and values of minimum of the Decision Cost Function (minDCF) for individual SR systems and their fusion. Best results (lowest error rates) are acquired when outputs of all the systems are fused. Note that this is true for both corpora NIST08 and NIST10.

	NIST08 EER [%]/minDCF	**NIST10** EER [%]/minDCF
GSV-NAP-256	7.27/0.0343	7.68/0.0393
GLDS-NAP-64	8.21/0.0365	9.16/0.0430
iVEC+PLDA	7.48/0.0376	8.74/0.0470
F-SVM	6.65/0.0311	7.05/0.0377
F-GSV-PLDA	6.49/0.0313	7.05/0.0383
F-GLDS-PLDA	6.60/0.0324	7.68/0.0391
F-ALL	**6.18/0.0300**	**6.74/0.0368**

[1]for details see http://www.itl.nist.gov/iad/mig/tests/spk/2008/sre08_evalplan_release4.pdf

[2]for details see http://www.itl.nist.gov/iad/mig/tests/spk/2010/NIST_SRE10_evalplan.r6.pdf

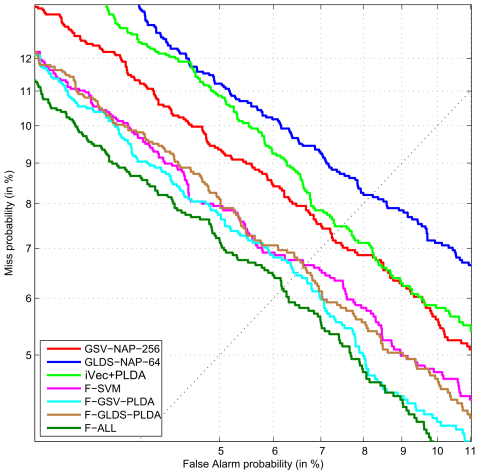

Fig. 1. DET curve depicting the dependency of missing a target speaker on the misclassification of a non-target speaker given a verification threshold. Results for various systems are obtained on trials from NIST SRE 2008.

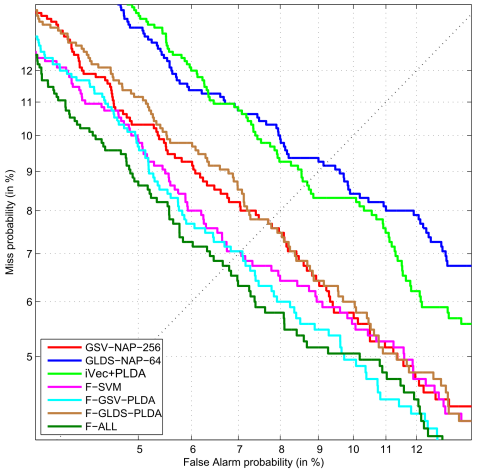

Fig. 2. DET curve depicting the dependency of missing a target speaker on the misclassification of a non-target speaker given a verification threshold. Results for various systems are obtained on trials from NIST SRE 2010. The fusion coefficients were trained on trials from NIST SRE 2008.

minDCF the cost of missing a target was set to 10, the cost of the false alarm was set to 1, and the probability of seeing a true trial was set to 0.01. These values are adopted from the NIST SRE 2008.

At first error rates for all three individual systems are given, F-SVM stands for the fusion of GLDS and GSV system, and F-ALL is the fusion of all three systems. Each fusion performs better than an individual system, but the lowest error rates are acquired when all three systems are fused. In Figure 1 and Figure 2 the Detection Error Trade-off (DET) curves are shown. Note that the fused system F-ALL behaves best along most of the curve, thus for each value of the verification threshold.

IX. Conclusions

Recent state-of-the-art methods used in the task of speaker recognition were presented. Complementarity of systems based on SVM models, GLDS, GSV supervectors, i-vectors extraction and PLDA modelling in the total variability space was analysed utilizing the linear logistic regression as a fusion tool. The decrease of error rates and increase in the performance of the speaker recognition system composed from mentioned subsystems was significant, hence the complementarity of reviewed techniques was proved.

Acknowledgments

This research was supported by the Grant Agency of the Czech Republic, project No. GAČR P103/12/G084.

References

[1] D. A. Reynolds, "A Gaussian Mixture Modelling Approach to Text-independent Speaker Identification," Ph.D. dissertation, Georgia Institute of Technology, 1992.

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

[3] 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," *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings*, vol. 1, pp. I–I, 2006.

[4] P. Kenny, "Joint Factor Analysis of Speaker and Session Variability: Theory and Algorithms," Centre de Recherche Informatique de Montréal (CRIM), Tech. Rep., 2006.

[5] N. Dehak, "Discriminative and Generative approaches for Long- and Short-term speaker characteristics modeling: application to speaker verification," Ph.D. dissertation, École de Technologie Supérieure, Université du Québec, 2009.

[6] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis For Speaker Verification," *IEEE Transactions on Audio, Speech and Language Processing*, 2010.

[7] S. J. D. Prince and J. H. Elder, "Probabilistic Linear Discriminant Analysis for Inferences About Identity," *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pp. 1–8, 2007.

[8] W. M. Campbell, "Generalized linear discriminant sequence kernels for speaker recognition," *IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP'02*, vol. 1, pp. I–161–I–164, 2002.

[9] K. Lee, C. You, T. Kinnunen, and D. Zhu, "Characterizing Speech Utterances for Speaker Verification with Sequence Kernel SVM," *Proceedings of Interspeech, Brisbane*, pp. 1397–1400, 2008.

[10] L. Machlica and Z. Zajíc, "Factor Analysis and Nuisance Attribute Projection Revisited," *Interspeech*, 2012.

[11] B. Schölkopf and A. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2001.

[12] R. Collobert, S. Bengio, and C. Williamson, "SVMTorch: Support vector machines for large-scale regression problems," *Journal of Machine Learning Research*, vol. 1, pp. 143–160, 2001.

[13] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A Study of Interspeaker Variability in Speaker Verification," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 16, pp. 980–988, 2008.

[14] N. Brümmer, "FoCal: Tools for fusion and calibration of automatic speaker detection systems," 2006. [Online]. Available: http://sites.google.com/site/nikobrummer/focal