Comparison of Generative and Discriminative Approaches for Speaker Recognition with Limited Data

Jan SILOVSKY, Petr CERVA, Jindrich ZDANSKY

SpeechLab, Technical University of Liberec, Studentska 2, 461 17 Liberec, Czechia

jan.silovsky@tul.cz, petr.cerva@tul.cz, jindrich.zdansky@tul.cz

Abstract. *This paper presents a comparison of three different speaker recognition methods deployed in a broadcast news processing system. We focus on how the generative and discriminative nature of these methods affects the speaker recognition framework and we also deal with intersession variability compensation techniques in more detail, which are of great interest in broadcast processing domain. Performed experiments are specific particularly for the very limited amount of data used for both speaker enrollment (typically ranging from 30 to 60 seconds) and recognition (typically ranging from 5 to 15 seconds). Our results show that the system based on Gaussian Mixture Models (GMMs) outperforms both systems based on Support Vector Machines (SVMs) but its drawback is higher computational cost.*

Keywords

Speaker recognition, Gaussian Mixture Models (GMM), Support Vector Machines (SVM), broadcast processing.

1. Introduction

The speaker recognition module is an essential part of any media data-mining system. Besides the straightforward benefit of information about who is speaking it also allows the speech recognition module to employ speaker-adapted acoustic models [1].

Speaker recognition techniques are traditionally based on generative classifiers like Gaussian Mixture Models (GMMs). However, recently discriminative classifiers represented particularly by Support Vector Machines (SVMs) [2] have been successfully applied to several fields of pattern recognition including speaker recognition. Discriminative classifiers are derived from statistical learning theory and have high generalization ability. Systems based on SVMs have done well in recent Speaker Recognition Evaluations (SREs) organized by the NIST [3] providing results comparable with state-of-the-art GMM based systems. The NIST SRE evaluations are conducted as a speaker verification task with defined sets of trials (test speaker versus utterance). The speaker recognition in broadcast streams states an open-set speaker identification task.

In this paper, we compare one system based on GMMs and two systems based on SVMs. The GMM-based system works directly with cepstral feature vectors and speaker models are derived from the Universal Background Model (UBM) by Maximum A Posteriori (MAP) adaptation [4], further we will refer to this system as to the UBM-GMM. The first SVM-based system (GMM-SVM) classifies feature vectors extracted as means of MAP adapted UBM [5]. Finally, the second SVM-based system (MLLR-SVM) uses the parameters derived by the speech recognizer for MLLR speaker adaptation as input for classification [6].

As one of the major sources of accuracy degradation in speaker recognition systems is a diversity of recording conditions and channels between sessions, experiments with an intersession variability compensation technique were carried out for all systems. The UBM-GMM system employed the eigenchannel adaptation [7], [8] which aims to adapt the speaker specific GMM trained under one channel conditions towards the different channel condition of test recording. Both SVM-based systems employed Nuisance Attribute Projection (NAP) [9]. The basic idea of the NAP is to project out dimensions that are irrelevant to the speaker recognition problem.

This paper provides full description of speaker recognition frameworks for both GMM and SVM based systems, deals with thorough evaluation of these systems and it also highlights some neglected implementation issues, which are of crucial importance for proper work of systems.

2. System Description

Let us assume that an acoustic segment was cut out of a broadcast stream by a segmentation routine e.g. that described in [10] and recognized as a speech by some acoustic classification routine. The task of a speaker recognition system is to decide whether the speaking person is one of the enrolled speakers, and if so which one, or that the speaker is unknown. All compared systems thus compound of a speaker identification and verification module.

Speaker identification module first provides the most probable identity hypothesis. This is subsequently passed to the speaker verification module which decides whether the voice of a speaker in the given segment belongs to the hypothesized speaker. The way of selection of the most probable speaker candidate and the way the verification is performed differ depending on the generative or discriminative nature of a system.

2.1 UBM-GMM System

The scheme of the UBM-GMM system is depicted in Fig. 1. For an F-dimensional feature vector $\mathbf{o}$, the Gaussian mixture density used for the likelihood function is defined as

$$P(\mathbf{o}|\lambda)=\sum_{c=1}^{C} w_c P_c(\mathbf{o}). \tag{1}$$

Hence the density is a weighted linear combination of C unimodal Gaussian densities $P_c(\mathbf{o})$. The λ represents speaker model parameterized by mixture weights w_c, mean vectors $\boldsymbol{\mu}_c$ and covariance matrices $\boldsymbol{\Sigma}_c$ (in general full, but most often and also in our case only diagonal), where $c = 1,\ldots,C$. The density $P_c(\mathbf{o})$ is defined as

$$P_c(\mathbf{o})=\frac{1}{\sqrt{(2\pi)^F \det \boldsymbol{\Sigma}_c}}\exp\left(-\frac{1}{2}(\mathbf{o}-\boldsymbol{\mu}_c)' \boldsymbol{\Sigma}_c^{-1}(\mathbf{o}-\boldsymbol{\mu}_c)\right). \tag{2}$$

Now let $\mathbf{O}=\{\mathbf{o}_1 \ldots \mathbf{o}_T\}$ be a sequence of feature vectors representing a parameterized signal. We suppose mutual independence of feature vectors of $\mathbf{O}$ and then we can compute the log-likelihood as

$$P(\mathbf{O}|\lambda)=\sum_{t=1}^{T} \log P(\mathbf{o}_t|\lambda). \tag{3}$$

When applying the maximum log-likelihood classifier, the enrolled speaker s^* is proclaimed as the originator of the recording according to

$$s^* = \arg\max_s P(\mathbf{O}|\lambda^s). \tag{4}$$

To speed up the identification process, the 10 top scoring Gaussian components[1] of the UBM were identified and stored for each frame of a recording and only these components were used while likelihood computation (1) for GMMs of enrolled speakers [4].

The decision whether to accept or reject the proposed identity s^* is based on the log-likelihood ratio test. The UBM is employed to represent acoustic space of imposters. Identity s^* is accepted if

$$\frac{1}{T}\left(P(\mathbf{O}|\lambda^{s^*})-P(\mathbf{O}|\lambda^{UBM})\right)>\theta \tag{5}$$

where θ is the verification threshold, otherwise is rejected.

[1] Unless stated otherwise, further we will refer to the Gaussian components just briefly as to the components.

The universal background model (UBM) is trained on data pooled from many background speakers. The models of enrolled speakers are derived from the UBM by classical Maximum A Posteriori (MAP) adaptation of the UBM means $\boldsymbol{\mu}_c$ using the following equation [4]

$$\boldsymbol{\mu}_c^s = \alpha_c E_c(\mathbf{O})+(1-\alpha_c)\boldsymbol{\mu}_c \tag{6}$$

where $c = 1,\ldots,C$. Here, $\mathbf{O}=\{\mathbf{o}_1 \ldots \mathbf{o}_T\}$ is a sequence representing the training utterance(s). The $E_c(\mathbf{O})$ is a new estimate of mean for component c given the observations $\mathbf{O}$. The α_c is the adaptation coefficient controlling the balance between old and new estimates. The α_c is defined as

$$\alpha_c = \frac{N_c}{N_c + r} \tag{7}$$

where r is the so called relevance factor, which is fixed for all components, and N_c is given by

$$N_c = \sum_t^T \gamma_c(t) \tag{8}$$

where, for each t, $\gamma_c(t)$ is the posterior probability of the event that the feature vector $\mathbf{o}_t$ was generated by the component c. Thus for $r \to 0$, the estimate of new parameters (potentially undertrained) is emphasized in (6), while the weight of the old parameters (better trained) emphasizes with the growing value of r.

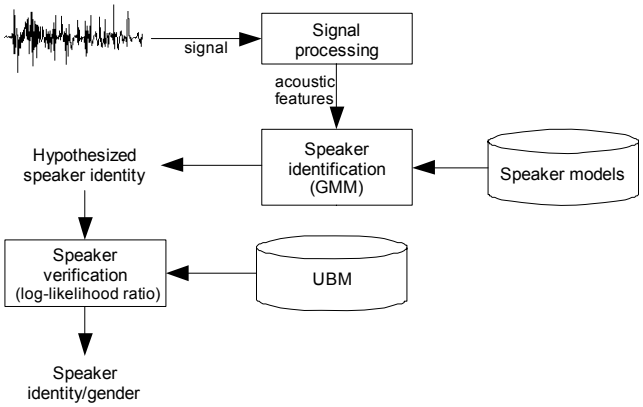

Fig. 1. Scheme of the UBM-GMM system.

2.2 GMM-SVM System

Support vector machine (SVM) [2] is a two-class linear classifier constructed from sums of a kernel function $K(.,.)$,

$$f(\mathbf{x})=\sum_{i=1}^{L} \alpha_i t_i K(\mathbf{x}, \mathbf{x}_i)+\xi \tag{9}$$

where the t_i are the ideal outputs (-1 or 1), $\sum_{i=1}^{L} \alpha_i t_i = 0$ and $\alpha_i > 0$. The vectors $\mathbf{x}_i$ are support vectors obtained from the training set by an optimization process [11]. For classification, a class decision is based upon whether the value $f(\mathbf{x})$ is above or below a decision threshold.

Fig. 2. shows the scheme of the GMM-SVM system. Mean vectors of MAP adapted UBM components are

stacked into one high-dimensional vector, referred to as the mean supervector, and used for classification by the SVM. Thus the supervector may be thought of as a mapping between an utterance of variable length and a vector of a fixed dimension.

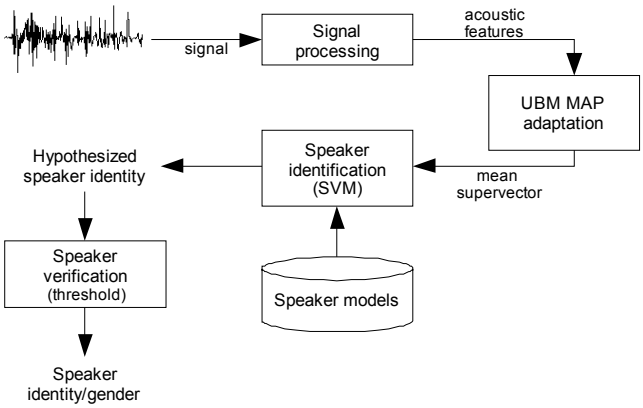

Fig. 2. Scheme of the GMM-SVM system.

We used the linear kernel derived based upon an approximation to Kullback-Leibler (KL) divergence between two GMM models [5]. The kernel function is defined as

$$K(\mathbf{O}_a,\mathbf{O}_b)=\sum_{c=1}^{C} w_c \boldsymbol{\mu}_c^a \boldsymbol{\Sigma}_c^{-1} \boldsymbol{\mu}_c^b = \sum_{c=1}^{C}\left(\sqrt{w_c}\,\boldsymbol{\Sigma}_c^{-1/2}\boldsymbol{\mu}_c^a\right)^T\left(\sqrt{w_c}\,\boldsymbol{\Sigma}_c^{-1/2}\boldsymbol{\mu}_c^b\right) \tag{10}$$

where w_c and $\sum_c$ are the weight and the covariance matrix of component c of the UBM. The mean vectors $\boldsymbol{\mu}_c^a$ and $\boldsymbol{\mu}_c^b$ were obtained by MAP adaptation of the UBM for utterance $\mathbf{O}_a$ and $\mathbf{O}_b$, respectively. (Note that as we adapt only mean vectors the weights and covariance matrices are same for all utterances).

A useful property of the kernel in (10) is that it allows utilization of the model compaction technique [5] and thus the SVM can be summarized as

$$f(\mathbf{x})=\left(\sum_{i=1}^{L}\alpha_i t_i b(\mathbf{x}_i)\right)^T b(\mathbf{x})+\xi=\mathbf{A}^T b(\mathbf{x})+\xi \tag{11}$$

where $b(\,)$ is the SVM expansion and $\mathbf{A}$ is the weighted sum of support vectors defining the SVM model. This means that a score for the target model and the GMM supervector is obtained by single inner product.

Since the SVM is a two-class classifier, we have to handle the speaker identification as a verification problem. The common method is *one vs. all* strategy when target speaker model is trained using positive samples represented by the speaker's data and negative samples are drawn from all other speakers enrolled in the system. However, strict following of this approach complicates progressive enrolment of new speakers, because with a new speaker, all existing models should be retrained using the complete set of all other speakers as impostors. We prefer speaker models to be independent of each other. Therefore the set of negative samples was drawn from background data used for the UBM training. These data are completely disjoint to the data of enrolled speakers. Identification is done in a *winner takes all* strategy, in which the classifier with the highest output function assigns the class (identity).

As the SVM itself normalizes the output score within a set of background speakers, no ratio is computed and the raw score is compared with a detection threshold for verification of the proclaimed speaker.

2.3 MLLR-SVM System

Both previous systems are primarily based on modeling of short-term cepstral features, but the problem of these features is that their distribution is not depending only on a speaker characteristic, but also on many other factors, particularly, environment properties, channel characteristics and the choice of words spoken. We will show methods attempting to cope with intersession variability for both GMM and SVM-based systems in next sections. A straightforward attempt to make models invariant to the choice of words is utilization of phone-constrained [12] or word-constrained [13] models. However, the most significant drawback in this case is the fragmentation of data which makes difficult estimation of well trained models.

Another approach proposed in [6] exploits the adaptation techniques employed by current speech recognition systems to turn the speaker-independent recognition model into more accurate speaker-dependent model. Speaker recognition with Maximum Likelihood Linear Regression (MLLR) transforms is based on modeling the difference between the speaker-dependent and speaker-independent models instead of modeling the cepstral observations directly. This difference is embodied in the coefficients of the affine transform and modeled as speaker features using SVMs. Scheme of the MLLR-SVM system is depicted in Fig. 3.

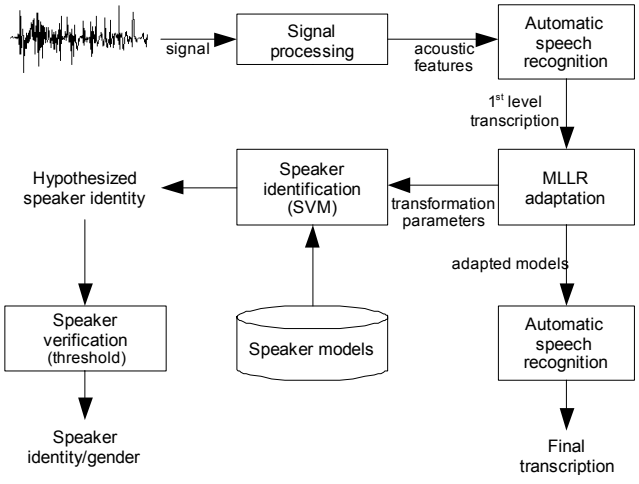

Fig. 3. Scheme of the MLLR-SVM system.

The MLLR adaptation module requires the knowledge of the transcription. Hence a speech recognition module has to precede it. From the perspective of the

speaker recognition system, this module causes relatively high increase of computational cost compared to the baseline system. However, as this step is anyway performed within two-stage speech recognition, there is no impact on the overall system performance.

In maximum likelihood linear regression (MLLR) [14], an affine transform is applied to the Gaussian mean vectors of *speech* recognition models to map from speaker-independent to speaker-dependent means as

$$\boldsymbol{\mu}' = \mathbf{A}\boldsymbol{\mu} + \mathbf{b} \tag{12}$$

where the adaptation parameters **A** and **b** are estimated so as to maximize the likelihood of the recognized speech under a preliminary recognition hypothesis. The features used for speaker classification are formed by concatenation of adaptation parameters into one high-dimensional vector.

2.4 Eigenchannel Adaptation

The eigenchannel adaptation technique [7], [8] copes with the intersession variability mismatch by adapting the GMM trained under one channel condition towards the different channel condition of a test recording. Keeping the notation used in previous description, let C be the number of GMM components and F the dimension of the feature vector. The CF dimensional mean supervector is formed by concatenation of GMM mean vectors. The supervector **m** is adapted to

$$\mathbf{m}_a = \mathbf{m} + \mathbf{V}\mathbf{x} \tag{13}$$

where **V** is an eigenchannel space matrix and **x** is a weight vector. The columns of **V** are commonly referred to as eigenchannels and they represent directions in which the most of intersession variability resides. Let R_c denote the number of eigenchannels. The components of **x**, referred to as channel factors, are obtained by maximizing the probability

$$p(\mathbf{O} \mid \mathbf{m} + \mathbf{V}\mathbf{x})\, p(\mathbf{x}) . \tag{14}$$

In estimating of the eigenchannel space matrix **V**, we followed the approach described in [15]. We have selected those speakers from the background dataset for which more than one recording was available. Let J denote the total number of recordings (in our case 1516). MAP adaptation of the UBM was performed separately for each speaker i and all his recordings $j = 1,\ldots,J_i$. Mean vectors of adapted GMMs were normalized by corresponding standard deviations and concatenated into the one supervector $\mathbf{s}_{i,j}$. Next, for each speaker, average supervector given by $\bar{\mathbf{s}}_i = \sum_{j=1}^{J_i} \mathbf{s}_{i,j} \big/ J_i$ was subtracted from each supervector so that $\mathbf{s'}_{i,j} = \mathbf{s}_{i,j} - \bar{\mathbf{s}}_i$. Finally, the supervectors $\mathbf{s'}_{i,j}$ formed columns of the $CF \times J$ matrix **S**. Eigenchannels **V** are then given by R_c eigenvectors of $CF \times CF$ within-speaker covariance matrix $\frac{1}{J}\mathbf{S}\mathbf{S}^T$ corresponding to the largest eigenvalues. The dimension CF is typically very large and direct computation of eigenvectors may be unfeasible. However, we can get these eigenvectors much more efficiently by calculating eigenvectors $\mathbf{V}'$ of the $J \times J$ matrix $\frac{1}{J}\mathbf{S}^T\mathbf{S}$. Eigenchannels are then given by $\mathbf{V} = \mathbf{S}\mathbf{V}'$. This product breaks the orthonormality of **V** and hence the length of eigenchannels must be normalized to one. Moreover, because of the short duration of our recordings, we found the MAP version of eigenchannel adaptation more appropriate than the ML version and in the MAP version the length of eigenchannels is normalized to the average within-speaker standard deviation of supervectors along the direction of the eigenchannel. More specifically, each column k of **V** is scaled by $\sqrt{2e_k}$, where e_k is the corresponding eigenvalue.

In the MAP version of eigenchannel adaptation, the channel factors vector **x** is considered to be normally distributed, i.e. the term $p(\mathbf{x})$ in (14) is substituted with the prior $\mathrm{N}(\mathbf{x};\mathbf{0},\mathbf{I})$ giving

$$p(\mathbf{O} \mid \mathbf{m} + \mathbf{V}\mathbf{x})\mathrm{N}(\mathbf{x};\mathbf{0},\mathbf{I}) . \tag{15}$$

The first step in estimating the channel factors **x** for a given test recording and a speaker model consists in calculation of the following Baum-Welch statistics for each Gaussian component c [15], [16]:

$$N_c = \sum_t^T \gamma_c(t) , \tag{16}$$

$$\mathbf{F}_c = \sum_t^T \left(\gamma_c(t)\left(\mathbf{o}(t) - \boldsymbol{\mu}_c\right)\right). \tag{17}$$

Let **N** be the $CF \times CF$ diagonal matrix whose diagonal blocks are $N_c\mathbf{I}(c = 1,\ldots,C)$, **F** be the $CF \times 1$ supervector obtained by concatenating $\mathbf{F}_c(c = 1,\ldots,C)$ and $\boldsymbol{\Sigma}$ be the $CF \times CF$ diagonal matrix whose diagonal blocks are $\boldsymbol{\Sigma}_c$ $(c = 1,...,C)$. Channel factors are then given by

$$\mathbf{x} = \left(\mathbf{I} + \mathbf{V}^T\mathbf{N}\mathbf{V}\right)^{-1}\mathbf{V}^T\boldsymbol{\Sigma}^{-½}\mathbf{F} \tag{18}$$

where $\mathbf{V}^T$ is the transpose of **V** and $\sum^{-½}$ denotes the inversed matrix $\sum$ with square root of variances (standard deviations) on the diagonal. If more than one speaker model is to be adapted (speaker identification case), a great speed-up is achieved by assuming a fixed occupation of the Gaussian mixture components and computing the $\gamma_c(t)$ probabilities only for the UBM. In this case, because the speaker models were derived by adaptation of means only and weights and covariance matrices are shared by all models, the only term that varies for different speakers in (18) is **F** and the product of the other terms may be precomputed only once for a test recording. Following the top N-best scoring components approach, for each frame only 10 probabilities $\gamma_c(t)$ estimated using the UBM were assumed to be non-zero, which allows efficient computation of **F**. Once channel factors are estimated, the mean supervector of the speaker model is adapted according to (13) (note that the supervector **m** is normalized by the corresponding standard deviations and the channel-adapted mean supervector $\mathbf{m}_a$ must be "denormalized" before likelihood computation using eq. 2).

2.5 Nuisance Attribute Projection

The basic idea of NAP is to remove dimensions that are irrelevant to the speaker recognition problem. The feature vectors **v** are transformed before they are passed to the SVM training process using the equation

$$\mathbf{v}_n = \mathbf{v} - \mathbf{E}\left(\mathbf{E}^T\mathbf{v}\right) \tag{19}$$

where **E** is the low-rank matrix defining the NAP subspace. NAP and eigenchannel adaptation techniques are in many aspects very similar [5]. The matrix **E** is equal to the matrix **V** in the ML version of eigenchannel adaptation and it is estimated by the same means.

Now let $b(\,)$ be the SVM expansion as in (10) and **P** be the projection defined by (19) (i.e. $\mathbf{P}=\mathbf{I}-\mathbf{E}\mathbf{E}^T$). The kernel with NAP is then given by

$$K\left(\mathbf{X}_a,\mathbf{X}_b\right)=\left[\mathbf{P}b(\boldsymbol{\mu}^a)\right]^T\left[\mathbf{P}b(\boldsymbol{\mu}^b)\right]. \tag{20}$$

As the matrix **E** is orthonormal ($\mathbf{E}^T\mathbf{E}=\mathbf{I}$), the projection **P** is idempotent ($\mathbf{P}^2=\mathbf{P}$) and the kernel can be rewritten as

$$K\left(\mathbf{X}_a,\mathbf{X}_b\right)=b(\boldsymbol{\mu}^a)^T\,\mathbf{P}b(\boldsymbol{\mu}^b)\,. \tag{21}$$

The transform must be applied to all supervectors before they are passed to the SVM model training; however it is not necessary to also transform the test supervectors before they are scored against the SVM models. Please note that although eq. 21 is not clearly unambiguous about the proper use of the NAP transform, as pointed out by [17], it is necessary to apply NAP transform before SVM training and it does not help to apply the NAP transform to test vectors or to models trained on the unprojected data.

2.6 SVM Feature Vector Scaling

As the SVM kernel is sensitive to the magnitude of the feature values, components must be scaled to avoid the values in greater numeric ranges dominate those in smaller numeric ranges. For the GMM-SVM system, such normalization is embodied directly in the SVM expansion. For the MLLR-SVM system, we found very useful to perform normalization of the SVM feature vector components by rank normalization (Rnorm). Rnorm replaces each feature value by its rank (normalized to the interval [0, 1]) in the background distribution. The side effect is that the original distribution is warped to approximately uniform distribution. Utilization of Rnorm implies application of NAP transform for both training and test data.

3. Experiments and Results

3.1 Datasets, Metrics and Evaluation Tasks Definition

Experiments were performed using our database of Czech BN streams. It contains mainly news streams collected in the period of more than five years. The whole captured streams were split into speaker homogeneous segments (of length usually ranging from 5 to 15 seconds).

The overall performance of a speaker recognition system was evaluated by the Recognition Error Rate R_E. For this metric, two following results were regarded as correct: a) correct gender was identified for a recording of a non-enrolled speaker; b) enrolled speaker was correctly identified and verified as the originator of a recording. This metric reflects well the experience of a user with a system. However, for proper system development, more detailed analysis is demanded. For instance, we are interested whether the discriminative nature of SVMs makes them more suitable for the speaker verification task, while utilization of generative models fits better the speaker identification task. Hence we are interested in the discrimination ability of both the speaker identification module and the speaker verification module separately. Further, development of a speaker verification module implies some calibration issues and a measure of how well is the system calibrated is also of our interest. Therefore three evaluation task were defined as follows:

- **closed-set speaker identification:** 228 enrolled speakers were chosen based on the amount of data available in the database. Models were trained for speakers with at least 30 seconds of speech data available for training and another 30 seconds for testing (regardless the number of segments). The data set preparation was done so that the training and test data for a particular speaker were from disjunct sessions (streams). The test data set contained 4245 recordings. This task was evaluated by the Closed-Set Identification Error Rate R_{CSE}.
- **speaker verification:** the test data set was extended by 2436 recordings from non-enrolled speakers giving 6681 recordings in total. Pre-evaluation versions of two systems (UBM-GMM and GMM-SVM) were used to find the best scoring models of enrolled speakers. Claimant speakers for a test recording were defined by top 5 speakers as identified by both systems (duplicities were discarded). This gives a total number of 49637 trials (4166 target trials). Speaker verification performance was measured by two metrics. The well-known Equal Error Rate (EER) reflects solely discrimination ability of evaluated systems, while the recently introduced [18], [19] log-likelihood-ratio cost function C_{llr} reflects both discrimination and calibration abilities. Chosen systems were compared via Detection Error Trade-off (DET) curves. Output scores were calibrated to be interpreted as a detection log-likelihood ratio. Linear mapping was found by the linear logistic regression using the FoCal toolkit[2].

[2] see http://www.dsp.sun.ac.za/~nbrummer/focal

Experiments were performed in a 2-fold cross-validation scenario with one fold used for calibration training and the second for testing, and vice versa.

- **open-set speaker identification:** the test data set is identical to the verification task. This means that 4245 recordings of the total number 6681 were produced by enrolled speakers. System performance for this task was measured by the R_E.

3.2 Common Signal Processing

All systems used classic Mel-frequency cepstral coefficient (MFCC) features. 13 MFCCs (including c0) were extracted from the signal and augmented with the first and second derivatives forming a 39-dimensional feature vector. Finally, Cepstral Mean Normalization (CMN) was applied.

3.3 Results of UBM-GMM

The three basic parameters which have impact on the performance of the UBM-GMM system are: a) the number of GMM components, b) the value of MAP relevance factor, and c) the number of eigenchannels. First we analyzed a relationship of the relevance factor value and the number of components. Authors usually conclude the relevance factor value to be irrelevant to the performance of speaker verification systems. However, our results, depicted in Fig. 4, show that the situation is not so straightforward. We see that the effect of the relevance factor value strengthens with the growing number of components. In a closer look, we found out that in the speaker verification task, the number of GMM components plays a crucial role and the relevance factor value is indeed rather irrelevant for all model sizes, while the performance in the closed-set speaker identification task is much more sensitive to a proper choice of the relevance factor value, particularly for larger models. Another observation is that the optimal value of relevance factor lowers with the growing model size.

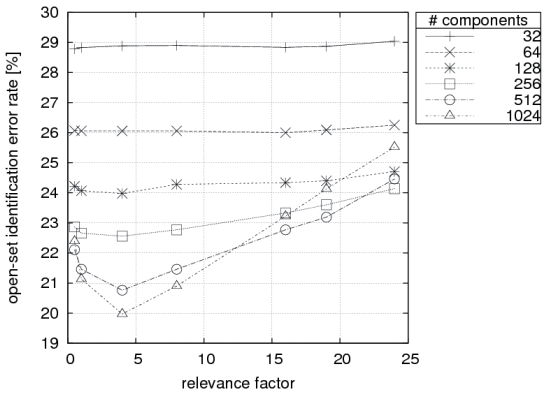

Fig. 4. The effect of the number of components and the relevance factor value on the UBM-GMM system.

The relevance factor values were chosen for models with 256, 512 and 1024 components based on the previous results and the systems with eigenchannel adaptation were evaluated with respect to the number of eigenchannels. Fig. 5 depicts the achieved results. Zero number of eigenchannels represents systems without eigenchannel adaptation.

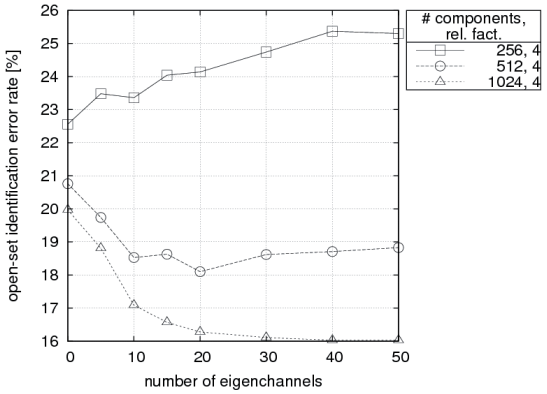

Fig. 5. The effect of eigenchannel adaptation for UBM-GMM systems with a non-optimal relevance factor value.

We see that eigenchannel adaptation clearly yields performance gain for system with 1024 and 512 components. For the system with 1024 components, the performance asymptotically increases with the number of eigenchannels, though there is only a marginal gain for more than 30 eigenchannels. This is expectable as an effect of normalizing the length of eigenchannels by corresponding eigenvalues (sorted in decreasing order). For the system with 512 components, we observe a slightly decreasing performance for more than 20 eigenchannels. This is quite surprising effect realizing that these "rare" eigenchannels are assumed to vary the eigenchannel adaptation only in a minor scope (thanks to the length normalization). We hypothesize that the models with 512 components provide less freedom, making it harder to separate the session and speaker variability. All session variability seems to be stacked in only 20 eigenchannels and next eigenchannels found by training algorithm probably reflect other factors. This hypothesis is further confirmed by results of the system with 256 components. Here the application of eigenchannel adaptation yields degradation of the performance regardless the number of eigenchannels.

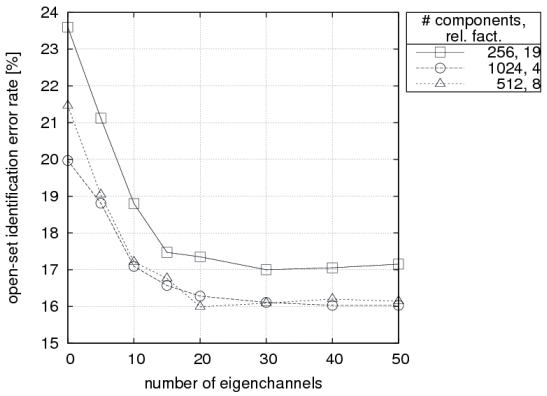

Fig. 6. The effect of eigenchannel adaptation for UBM-GMM systems with the optimal relevance factor value.

Based on the results of previous experiments and taking into account the limited amount of data available for speaker enrolment and testing, we decided to increase the value of the relevance factor for systems with smaller number of components in order to provide more freedom to the MAP adaptation process. Fig. 6 summarizes the achieved results.

Remarkable improvement of performance is apparent for both systems with 256 and 512 components. The system with 512 components now yields the same performance as the system with 1024 components, which allows significant speed-up of the recognition process. The system with 512 components saves up to 60 % of computational cost compared to the system with 1024 components (both with 30 eigenchannels).

3.4 Results of GMM-SVM

Likewise for the UBM-GMM system, the three basic parameters which are supposed to affect the performance of the GMM-SVM system are: a) the number of GMM components, b) the value of MAP relevance factor, and c) the dimension of NAP subspace (analogously to the number of eigenchannels). First, we again analyzed the effect of the number of GMM components and the relevance factor value for the system without intersession variability compensation. There is no reason for the performance drop in the case of utilization of too large GMM models in the UBM-GMM system as the unaltered components (due to the insufficient amount of data) stay away of the likelihood computation. On the contrary, in the GMM-SVM system, these unaltered components form the feature vector as well as the other components and thus corrupt the classification. The choice of the number of GMM components is hence of a crucial importance as depicted in Fig. 7. The relevance factor value has also a substantial impact on the performance; however, the lowest evaluated value (1.0) yielded the best performance for all model sizes.

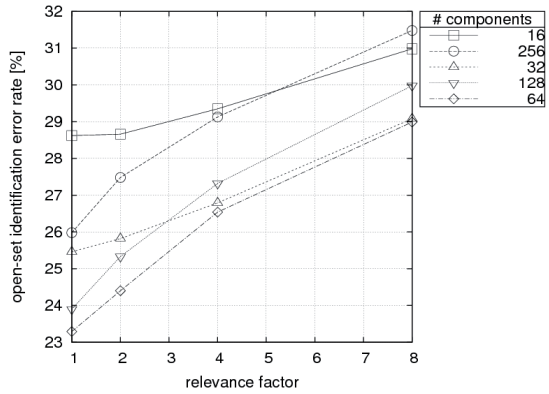

Fig. 7. The effect of the number of components and the relevance factor value on the GMM-SVM system.

Fig. 8 illustrates the performance of systems with nuisance attribute projection employed for intersession variability compensation. Application of the NAP has no effect on the optimal value of relevance factor for a system with particular number of GMM components. NAP yields notable improvement of the performance, however in a slightly lower scope compared to the eigenchannel adaptation effect on UBM-GMM systems. This is probably caused due to the different principle of eigenchannel adaptation and NAP. While NAP only blindly projects out the channel effects using fixed transformation of the feature vector, eigenchannel adaptation shifts the speaker model towards the channel conditions of a test recording based on the maximum likelihood criterion. We also carried out experiments with application of Rnorm for GMM-SVM systems and concluded Rnorm to be of no use.

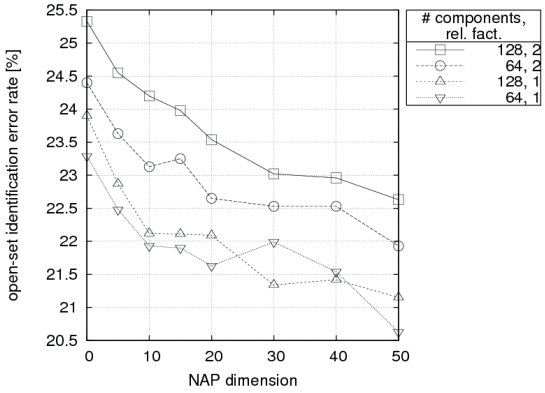

Fig. 8. The effect of nuisance attribute projection on the GMM-SVM system.

3.5 Results of MLLR-SVM

Deployment of the MLLR-SVM system within the limited data task is much more challenging compared to the both previous systems since robust estimation of MLLR transformation is highly dependent on the amount of data available. Initially, we tested two basic variants of the MLLR-SVM system differing in the number of regression classes. The first variant used single global transformation for all Gaussians in the HMM. The second variant employed more detailed adaptation scenario with three regression classes. HMM Gaussians were clustered into these classes by their similarity (data-driven approach). In both cases, the non-speech units in the recording were left out of adaptation process since they are not expected to help in speaker recognition. A linear inner-product kernel function was used for both variants. The size of the SVM feature vector for the first system was 1560 (39-dimensional feature vector) and for the second system 4680. We do not report results for the later system with 3 regression classes since it yielded remarkably worse performance. This indicates that the available amount of data is insufficient for estimation of too many parameters.

Our speech recognizer operates with gender specific HMMs and the MLLR transforms are hence dependent on the chosen gender model. Misclassification of speaker's gender for a test recording is rather rare; however, it leads to a notable drop of average system performance. A possible solution how to avoid problems with gender recogni-

tion is utilization of transforms derived for both gender specific models (male and female) combined into one larger feature vector [20]. Moreover, we can expect further performance improvement since gender specific HMMs are not just linear transforms of each other and they provide two different views of the observation space. The size of the SVM feature vector was 3120 in this case.

Fig. 9. summarizes achieved results. We highlight the effect of the Rnorm which yielded substantial improvement of the performance. Results also confirm benefit of utilization of transforms estimated for both gender dependent models. MLLR transform features are supposed to be invariant to the channel effects and, indeed, application of the NAP had only a minor effect on the performance of systems (particularly those with the Rnorm). Finally, we have to conclude that the performance is substantially worse compared to the results of the other evaluated systems. We will discuss the source of the performance drop further in the next section.

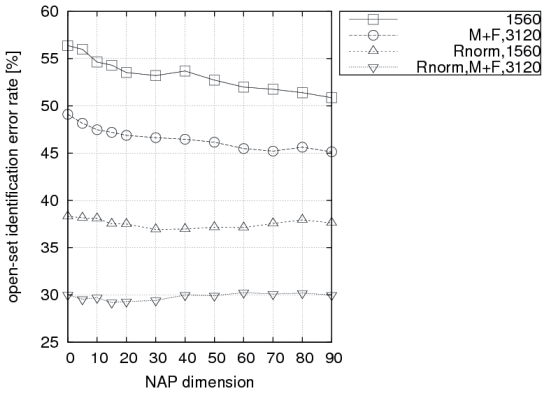

Fig. 9. Results of MLLR-SVM systems.

3.6 Comparison of Systems

Tab. 1 shows evaluated metrics and execution time required to process the evaluation data (as a multiple of real-time, extraction of MFCC features from signal is not included) for best performing systems. Fig. 10 illustrates the difference in the discriminative ability of the systems in the speaker verification task using the DET curves. We conclude that UBM-GMM systems provide superior performance in both speaker identification and verification tasks. The performance drop of SVM-based systems in the speaker verification is substantially lower compared to the drop in the closed-set speaker identification task. The overall performance is subsequently affected by the poor performance in the closed-set speaker identification task as the speaker verification module is not able to compensate for the incorrectly recognized speaker identity. This proves that classifiers with discriminative nature are much more suitable for the speaker verification task than the speaker identification task.

A great advantage of GMM-SVM systems is their speed. There are several reasons why GMM-SVM systems are much faster than UBM-GMM systems: a) GMM-SVM systems use models with less components, b) scoring of a SVM model consists only of a computation of the dot product of vectors compared to the more complex likelihood computation for MAP adapted GMM models (even though only a limited number of top components is scored for each frame), c) the computational cost of eigenchannel adaptation significantly raises with the growing number of eigenchannels as the eigenchannel factors are estimated using maximum likelihood computation while the NAP has no effect on the overall computational cost of GMM-SVM systems (regardless the NAP dimension.)

	R_{CSE}	EER	C_{llr}	R_E	x RT
UBM-GMM					
512 c, r=8, channs=20	7.44	6.96	0.260	15.99	0.158
1024 c, r=4, channs=30	8.22	6.91	0.254	16.11	0.546
GMM-SVM					
64 c, r=1, channs=50	10.84	7.61	0.276	20.63	0.007
128 c, r=1, channs=30	11.10	8.40	0.301	21.34	0.016
MLLR-SVM					
M+F, Rnorm, channs=0	17.49	9.28	0.328	30.01	-

Tab. 1. The comparison of evaluated metrics for best performing systems.

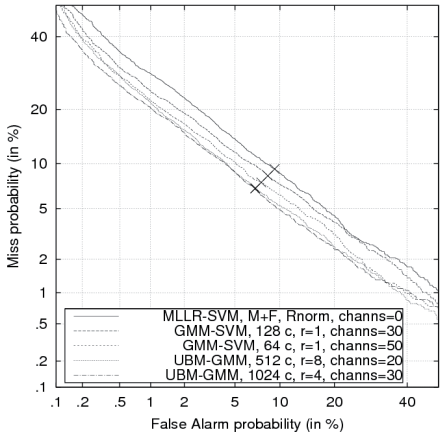

Fig. 10. DET curves comparison for best performing systems.

MLLR-SVM systems performed worst in all evaluated metrics. A possible reason may be short duration of test recordings which disallows robust estimation of MLLR transform features. We find support for this hypothesis in the results achieved for the system with three regression classes. We do not report the execution time for the MLLR-SVM system in Tab. 1 since the time required for scoring of SVM models against feature vectors derived for test recordings was negligible (< 0.001 x RT). This would suggest that this system is the fastest in our evaluation. However, this holds only if we take MLLR transforms as by-product of a speech recognition system with MLLR speaker adaptation. When we use feature vectors composed of transforms derived for both male and female models we have to perform additional calculation of MLLR transforms and this immoderately increases the computational cost.

Finally, two commonly used score-level normalization techniques were examined, namely Znorm and Tnorm. However, no improvement of accuracy was observed for neither of these normalization techniques.

4. Conclusions and Future Work

This paper dealt with thorough analysis and comparison of three speaker recognition systems in the domain of broadcast news processing with limited amount of data. The UBM-GMM system represented the generative approach and the discriminative approach was represented by the GMM-SVM system and MLLR-SVM system. First, we analyzed and discussed the effect of various parameters of particular systems. We pointed out that some parameters, commonly stated to be irrelevant to the system performance, may play a crucial role in systems dealing with limited data. We also demonstrated the importance of utilization of intersession variability compensation techniques. Subsequently, we compared the results of best performing systems. The GMM-based system outperformed both discriminative based systems. The performance of the SVM-based systems is corrupted particularly by low accuracy in the closed-set speaker identification task. Computational effectiveness of systems was also discussed and the low computational cost of the GMM-SVM highlighted. In the future work, we will focus on the fusion of evaluated systems, particularly in the task of speaker verification. The best of the proposed method is going to be applied in the system for automatic broadcast program transcription and information retrieval [21].

Acknowledgements

The research was supported by Czech Science Foundation (GACR) grants no. 102/07/P430 and 102/08/0707.

References

[1] CERVA, P., NOUZA, J., SILOVSKY, J. Two-step unsupervised speaker adaptation based on speaker and gender recognition and HMM combination. In *Interspeech 2006*. Pittsburg (USA), September, 2006.

[2] BURGES, C. J. C. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 1998, vol. 2, no. 2, pp. 1–47.

[3] The NIST Year 2005 Speaker Recognition Evaluation Plan. http://www.nist.gov/speech/tests/sre/2005/sre-05_evalplan-v6.pdf

[4] REYNOLDS, D. A., QUATIERI, T. F., DUNN, R. Speaker verification using adapted gaussian mixture models. *Digital Signal Processing*, 2000, vol. 10, no. 1-3, pp. 19-41.

[5] CAMPBELL, W. M., STURIM, D. E., REYNOLDS, D. A., SOLOMONOFF, A. SVM-based speaker verification using a GMM supervector kernel and NAP variability compensation. In *Proc. ICASSP*. 2006, pp. 97-100.

[6] STOLCKE, A., FERRER, L., KAJAREKAR, S., SHRIBERG, E., VENKATARAMAN, A. MLLR transforms as features in speaker recognition. In *Proc. Interspeech*. Lisbon, Sep. 2005, pp. 2425-2428.

[7] KENNY, P., DUMOUCHEL, P. Disentangling speaker and channel effects in speaker verification. In *Proc. ICASSP*. Montreal (Canada), May 2004, vol. 1, pp. 47.40.

[8] VOGT, R., SRIDHARAN, S. Experiments in session variability modelling for speaker verification. In *Proc. ICASSP*. Toulouse (France), May 2006, vol. 1, pp. 897-900.

[9] SOLOMONOFF, A., CAMPBELL, W. M., BOARDMAN, I. Advances in channel compensation for SVM speaker recognition. In *Proc. ICASSP*. Philadelphia (PA, USA), Mar. 2005, vol. I, pp. 629-632.

[10] ZDANSKY, J., NOUZA, J. Detection of acoustic change-points in audio records via global BIC maximization and dynamic programming. In *Interspeech 2005*. Lisboa (Portugal), September, 2005, pp. 669-672, ISSN 1018-4074.

[11] CHANG, Ch. Ch., LIN, Ch. J. LIBSVM: a library for support vector machines. 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

[12] PARK, A., HAZEN, T. J. ASR dependent techniques for speaker identification. In *Proc. ICSLP*, J. H. L. Hansen and B. Pellom, Eds., Denver, Sept. 2002, pp. 13371340.

[13] STURIM, D. E., REYNOLDS, D. A., DUNN, R. B., QUATIERI, T. F. Speaker verification using text-constrained gaussian mixture models. In *Proc. ICASSP*. Orlando (Florida), May 2002, IEEE, pp. I:677680, 1317.

[14] GALES, M. J. F. Maximum likelihood linear transformations for HMM-based speech recognition. *Computer Speech and Language*, 1998, vol. 12, no. 2, pp. 75-98.

[15] BURGET, L., MATEJKA, P., SCHWARZ, P., GLEMBEK, O., CERNOCKY, J. Analysis of feature extraction and channel compensation in a GMM speaker recognition system. In *Proc. ICASSP*. Honolulu (USA), 2007, vol. 15, pp. 1979-1986.

[16] KENNY, P., BOULIANNE, G., OUELLET, P., DUMOUCHEL, P. Joint factor analysis versus eigenchannels in speaker recognition. *IEEE Trans. Audio, Speech, Lang. Process.*, May 2007, vol. 15, no. 4, pp. 1435–1447.

[17] BRUMMER, N., BURGET, L., CERNOCKY, J., GLEMBEK, O., GREZL, F., KARAFIAT, M., VAN LEEUWEN, D. A., MATEJKA, P., SCHWARZ, P., STRASHEIM, A. Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006. In *Proc. ICASSP*. Honolulu (USA), 2007, vol. 15, pp. 2072-2084.

[18] BRUMMER, N., DU PREEZ, J. Application-independent evaluation of speaker detection. *Computer Speech and Language*, 2006, vol. 20, pp. 230-275.

[19] VAN LEEUWEN, D. A., BRUMMER, N. An introduction to application-independent evaluation of speaker recognition systems. In *Lecture Notes in Artificial Intelligence*, pp. 330-353, 2007.

[20] STOLCKE, A., FERRER, L., KAJAREKAR, S. Improvements in MLLR-transform-based speaker recognition. In *Proc. IEEE Odyssey 2006 Speaker and Language Recognition Workshop*. San Juan (Puerto Rico), June 2006, pp. 16.

[21] NOUZA, J., ZDANSKY, J., CERVA, P., KOLORENC, J. A system for information retrieval from large records of broadcast programs. In *Text, Speech and Dialogue. Lecture Notes in Artificial Intelligence*. Berlin: Springer-Verlag, 2006, pp. 401-408.

About Authors...

Jan SILOVSKY (1982) received the Master degree at the Technical University of Liberec (TUL) in 2006. He is currently a PhD student at the Institute of Information Technology and Electronics TUL. His research work is focused on speaker and speech recognition.

Petr CERVA (1980) received the Master degree (2004) and Ph.D. degree (2007) at the Technical University of Liberec (TUL). Since 2007 he has been at the Technical University of Liberec (TUL) as an assistant professor. His research work is focused on speaker adaptation and speech recognition.

Jindrich ZDANSKY (1978) received the Master degree in Electronics and Electronic Systems from CTU Prague and the Ph.D. degree in Technical Cybernetics from TU Liberec, in 2002 and 2005, respectively. He is currently an assistant professor at the Institute of Information Technology and Electronics TUL. His research interests are speaker-change detection and speech recognition.

RADIOENGINEERING REVIEWERS I

September 2009, Volume 18, Number 3

- ADALAN, A., Vienna University of Technology, Austria
- ALA-LAURINAHO, J., Helsinki University of Technology, Finland
- AVENDANO, C., Creative Advanced Technology Center, CA, USA
- BALLING, P., Antenna Systems Consulting ApS, Denmark
- BEZOUŠEK, P., University of Pardubice, Czechia
- BIOLEK, D., University of Defense, Brno, Czechia
- BONEFAČIĆ, D., University of Zagreb, Croatia
- BRANČÍK, L., Brno University of Technology, Czechia
- CAPPELINI, V., Universita di Firenze, Italy
- CIMINO, M., University of Bordeaux, France
- COLLADO, A., Centre Tecnologic de Telecomunicacions de Catalunya, Barcelona, Spain
- ČERMÁK, D., University of Pardubice, Czechia
- ČERNOCKÝ, J., Brno Univ. of Technology, Czechia
- DJIGAN, V., R&D Center of Microelectronics, Russia
- DOBEŠ, J., Czech Technical University in Prague, Czechia
- DOBOŠ, L., Technical University of Košice, Slovakia
- DOSTÁL, T., Brno University of Technology, Czechia
- DŘÍNOVSKÝ, J., Brno University of Technology, Czechia
- FEDRA, Z., Brno University of Technology, Czechia
- FRÝZA, T., Brno University of Technology, Czechia
- GAJDOŠÍK, L., VŠB - Technical University of Ostrava, Czechia
- HANUS, S., Brno University of Technology, Czechia
- HAZDRA, P., Czech Technical University in Prague, Czechia
- HÁJEK, K., University of Defence, Brno, Czechia
- HORSKÝ, P., AMIS Czech, Ltd., Brno, Czechia
- HOZMAN, J., Czech Technical University in Prague, Czechia
- HSIEN CHU-WU, National Taichung Institute of Technology, Taichung, Taiwan
- JUHÁR, J., Technical University of Košice, Slovakia
- KASAL, M., Brno University of Technology, Czechia
- KEJÍK, P., Brno University of Technology, Czechia
- KESKIN, A. U., Yeditepe University, Istanbul, Turkey
- KI YOUNG KIM, National Cheng Kung University, Tainan, Taiwan
- KOLÁŘ, R., Brno University of Technology, Czechia
- KOLKA, Z., Brno University of Technology, Czechia
- KOVÁCS, P., Brno University of Technology, Czechia