Study on the Relevance Factor of Maximum *a Posteriori* with GMM for Language Recognition

Chang Huai You, Haizhou Li, Kong Aik Lee

Institute for Infocomm Research (I^{2}R), A*STAR, Singapore

{echyou,hli,kalee}@i2r.a-star.edu.sg

Abstract

In this paper, the relevance factor in maximum *a posteriori* (MAP) adaptation of Gaussian mixture model (GMM) from universal background model (UBM) is studied for language recognition. In conventional MAP, relevance factor is typically set as a constant empirically. Knowing that relevance factor determines how much the observed training data influence the model adaptation, thus the resulting GMM models, we believe that the relevance factor should be dependent to the data for more effective modeling. We formulate the estimation of relevance factor in a systematic manner and study its role in characterizing spoken languages with supervectors. We use a Bhattacharyya-based language recognition system on National Institute of Standards and Technology (NIST) language recognition evaluation (LRE) 2009 task to investigate the validate of the data-dependent relevance factor. Experimental results show that we achieve improved performance by using the proposed relevance factor.

Index Terms: maximum *a posteriori*, supervector, Gaussian mixture model, support vector machine

10.21437/Interspeech.2011-724

1. Introduction

Language recognition is the process of recognizing the language of a spoken utterance. Common techniques used in language recognition include the acoustic and phonotactic modeling approaches. The parallel-phone recognition followed by language modeling (PPR-LM) [1] is a typical phonotactic approach that classifies spoken languages by using statistics over phone tokens. Another effective alternative is Gaussian mixture model (GMM) that relies solely on acoustic features. Recently, acoustic approaches such as GMM supervector achieve state-of-the-art performance.

In this paper, we continue the study on GMM supervector for language recognition, one of the most popular acoustic modeling approaches for its reliable performance [2]. In GMM approach, a language model is obtained by maximum *a posteriori* (MAP) estimation from a universal background model (UBM) [3]. The UBM is usually trained through expectation-maximization (EM) algorithm from a background dataset covering a wide range of languages, speakers and channels. In MAP, the relevance factor is indirectly controlling how much new data could affect the updating of parameters (i.e., weight, mean, covariance). It is believed that the relevance factor can also be optimized by the particular background data. Conventional MAP does not define the relevance factor in a systematic manner; in other words, the relevance factor is usually set empirically. Most of researchers like to use an appropriate fix value in place of the data-dependent value. In the GMM-UBM system, the relevance factor is less sensitive and therefore can be fixed. This is possibly due to the nature of generative modeling [4]. However, support vector machine (SVM) works in a discriminative manner. In GMM-SVM language recognition system, a GMM supervector is used to represent the language property of an utterance and serves as an input vector to the SVM. It is important to reduce the variation of database so that supervectors can well manifest the saliency of language characteristics.

Since our discussion focuses on the GMM supervector rather than the GMM probability, it is straightforward to approach the problem in the supervector domain. It is observed that the supervector deduced from the MAP criterion can be also derived in supervector domain through the probabilistic analysis [5]. In [6], we have shown the effectiveness of the adaptation of the relevance factor to the duration of the particular utterance for language recognition. In this paper, we focus on the mathematical derivation of the data-dependent relevance factor in connection with the universal background database, where we analyze the relevance factor of MAP through the supervector modeling and derive specifically the algorithm to obtain the relevance factor as well as the related parameters. Actually, the mathematical derivation of the relevance factor of MAP in this paper can be viewed as a special case of joint factor analysis (JFA) in [5] where the eigen-channel and eigen-voice factors are considered in the mathematic derivation targeted for speaker recognition. In this paper, we mainly emphasize the mathematic derivation of classical MAP in supervector domain and give a clear analysis of MAP and the estimation of the relevance factor.

In the SVM framework, we need to define a kernel to compare supervectors for classification. We implement a language recognition system based on the Bhattacharyya-based kernel [7]. The validity of the data-dependent relevance factor will be investigated by using the language recognition system on the NIST LRE 2009 core tasks [8].

In the remainder of the paper, we describe the conventional MAP for GMM in section 2. We derive the relevance factor for MAP estimation in section 3. The performance evaluation is reported in section 4. We summarize the paper in section 5.

2. MAP for language recognition

Usually, with EM algorithm, the UBM is trained using a large dataset to form a language-independent model [4]. The selection of dataset has to consider different languages, channels and speakers. The UBM can be denoted as

$$\mathrm{u} = \{\omega_i, \mathbf{m}_i, \Sigma_i; i = 1, 2, ..., C\} \quad (1)$$

while the language-dependent GMM, λ, has the same form

$$\lambda = \{\hat{\omega}_i(\lambda), \hat{\mathbf{M}}_i(\lambda), \hat{\Sigma}_i(\lambda); i = 1, 2, ..., C\} \quad (2)$$

where $\mathbf{m}_i$ (or $\mathbf{M}_i$), Σ_i, ω_i, $(i = 1, ..., M)$ are respectively the mean vector, the covariance matrix, and the weight of the ith Gaussian component.

For the MAP adaptation to λ, prior knowledge is given by using the prior distribution over λ, $P(\lambda)$. With the MAP criterion, λ is selected such that it maximizes the *a posteriori* probability,

$$\lambda = \arg\max_{\lambda} P(\lambda|\mathbf{X}) = \arg\max_{\lambda} \left[p(\mathbf{X}|\lambda)P(\lambda)\right] \quad (3)$$

where $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_\kappa]$ is the sequence of feature vectors used to train the GMM, λ; $\mathbf{x}$ is a J-dimensional feature vector; and κ is the number of feature vectors. As a result of (3), the mean parameters of the ith Gaussian are adapted as follows [4],

$$\mathbf{M}_i(\lambda, j) = \alpha_i(j)\Xi_i(\lambda, j) + (1 - \alpha_i(j))\mathbf{m}_i(j) \quad (4)$$

where Ξ_i is the first order sufficient statistics. $\alpha_i(j)$ $(j = 1, ..., J)$ are the data-dependent adaptation coefficients, which are given by

$$\alpha_i(j) = \frac{N_i}{N_i + \gamma_i(j)} \quad (5)$$

The relevance factor γ_i is the parameter in the normal-Wishart density as which the Gaussian parameters are modeled. However, in conventional MAP, the relevance factor is given as a fixed value, and the occupation rate N_i is theoretically given by

$$N_i = \sum_{t=1}^{\kappa} \frac{\omega_i p(\mathbf{x}_t|\mathbf{m}_i, \Sigma_i)}{\sum_{l=1}^{C} \omega_l p(\mathbf{x}_t|\mathbf{m}_l, \Sigma_l)} \quad (6)$$

where $p(\cdot)$ denotes probabilistic function.

Using a data-dependent adaptation coefficient allows a mixture-dependent adaptation of parameters. If a mixture component has a low occupation rate N_i of new data, then $\alpha_i(j) \to 0$ causing the deemphasis of the new parameters. For mixture components with high probabilistic counts, $\alpha_i(j) \to 1$, causing the use of the new language-dependent parameters. It is obviously found that the relevance factor is a way of controlling how much new data should be observed in a mixture before the new parameters begin replacing the old parameters. Thus, this approach should be robust to sparse training data.

3. Relevance factor

In Bayesian statistics, maximum *a posteriori* (MAP) is a way to estimate the parameters of the probabilistic distribution. The MAP estimation can be realized in various ways, for examples, numerical optimization with first and second derivatives, modification of an expectation-maximization algorithm and a Monte Carlo method etc. Conventionally, when MAP method is used for GMM parameter estimation, its relevance factor is considered as fixed value, which is believed not to be an optimal solution. In order to meet the realistic requirement of the GMM-supervector based description for the language recognition, we have to analyze the influence to the relevance factor and to provide a proper solution in the MAP algorithm. It is known that the relevance factor plays a different role in different application. In this session, we are analyzing and showing the relationship between relevance factor and the property of the training data.

3.1. Determination of relevance factor

We observed that the supervector deduced from the MAP criterion can be also derived in supervector domain through the probabilistic analysis. We analyze the MAP algorithm from the GMM-supervector perspective. In language recognition, the GMM-supervector is usually generated from UBM to represent the language characteristics according to the related utterances. Here we define the supervector as a concatenation of mean vectors from a GMM. Assume $\mathbf{m}$ represents the UBM supervector and also assume GMM-supervector $\mathbf{M}(\lambda)$ can be constructed by a language independent vector $\mathbf{m}$ and a language dependent vector $\tilde{\mathbf{m}}(\lambda) = \Phi\mathbf{z}(\lambda)$, where Φ denotes a transit matrix reflecting some feature of generalized training database and vector $\mathbf{z}(\lambda)$ is related to certain attributes of the particular language. We have

$$\mathbf{M}(\lambda) = \mathbf{m} + \Phi\mathbf{z}(\lambda) \quad (7)$$

It is reasonable to assume that Gaussian components in a GMM are independent each other; and further assumption is that the language-dependent vector $\mathbf{z}(\lambda)$ is of the standard normal distribution $\aleph(\mathbf{z}(\lambda)|0, 1)$, and Φ is a block diagonal matrix with each block being of dimension $J \times J$, hence the mean vector of the ith Gaussian component can be given by

$$\mathbf{M}_i(\lambda) = \mathbf{m}_i + \Phi_i\mathbf{z}_i(\lambda) \quad (8)$$

the natural logarithm of the conditional likelihood function of an observed feature vector $\mathbf{x}$ given the attribute $\mathbf{z}(\lambda)$ is shown below

$$\log P_\lambda(\mathbf{x}|\mathbf{z}(\lambda)) = \Theta(\lambda) + \Omega(\lambda, \mathbf{z}(\lambda)) \quad (9)$$

where $\Theta(\lambda)$ accounts for all terms unrelated to $\mathbf{z}(\lambda)$

$$\Theta(\lambda) = \sum_{i=1}^{C} N_i(\lambda) \log \frac{1}{(2\pi)^{J/2}|\Sigma_i(\lambda)|^{1/2}} - tr(\Sigma^{-1}(\lambda)S(\lambda, \mathbf{m})) \quad (10)$$

where $tr(\cdot)$ denotes the trace of matrix. $\Omega(\lambda, \mathbf{z}(\lambda))$ encompasses all terms related to $\mathbf{z}(\lambda)$, i.e.

$$\Omega(\lambda, \mathbf{z}(\lambda)) = \mathbf{z}^*(\lambda)\Phi^*\Sigma^{-1}(\lambda)\Xi(\lambda, \mathbf{m}) - \frac{1}{2}\mathbf{z}^*(\lambda)\Phi^* N(\lambda)\Sigma^{-1}(\lambda)\Phi\mathbf{z}(\lambda) \quad (11)$$

actually the occupation rate N and the first order statistics Ξ depend on λ, and $\Xi(\lambda, \mathbf{m}) = \begin{pmatrix} \Xi(\lambda, \mathbf{m}_1) \\ \cdots \\ \Xi(\lambda, \mathbf{m}_C) \end{pmatrix}$ where $\Xi(\lambda, \mathbf{m}_i) = \sum_t (\mathbf{x}_t - \mathbf{m}_i)$; and $S(\lambda, \mathbf{m})$ is the second order statistics. Eq. (9) can be proven as follows:

$$\log P_\lambda(\mathbf{x}|\mathbf{z}(\lambda)) = \sum_{i=1}^{C} N_i(\lambda) \log \frac{1}{(2\pi)^{J/2}|\Sigma_i(\lambda)|^{1/2}} - \frac{1}{2}\sum_{i=1}^{C}\sum_{t}(\mathbf{x}_t - \mathbf{M}_i(\lambda))^*\Sigma_i^{-1}(\lambda)(\mathbf{x}_t - \mathbf{M}_i(\lambda)) \quad (12)$$

The second term of (12) can be expanded and simplified as follows

$$\begin{aligned}
&\sum_{i=1}^{C}\sum_{t}(\mathbf{x}_t-\mathbf{M}_i(\lambda))^*\Sigma_i^{-1}(\lambda)(\mathbf{x}_t-\mathbf{M}_i(\lambda))\\
&=\sum_{i=1}^{C}\sum_{t}(\mathbf{x}_t-\mathbf{m}_i)^*\Sigma_i^{-1}(\lambda)(\mathbf{x}_t-\mathbf{m}_i)\\
&\qquad-2\sum_{i=1}^{C}\sum_{t}O_i^*(\lambda)\Sigma_i^{-1}(\lambda)(\mathbf{x}_t-\mathbf{m}_i)\\
&\qquad+\sum_{i=1}^{C}O_i^*(\lambda)N_i(\lambda)\Sigma_i^{-1}(\lambda)O_i(\lambda)\\
&=\sum_{i=1}^{C}tr(\Sigma_i^{-1}(\lambda)S_i(\lambda,\mathbf{m}_i))-2\sum_{i=1}^{C}O_i^*(\lambda)\Sigma_i^{-1}(\lambda)\Xi_i(\lambda,\mathbf{m}_i)\\
&\qquad+\sum_{i=1}^{C}O_i^*(\lambda)N_i(\lambda)\Sigma_i^{-1}(\lambda)O_i(\lambda)\\
&=tr(\Sigma^{-1}(\lambda)S(\lambda,\mathbf{m}))-2O^*(\lambda)\Sigma^{-1}(\lambda)\Xi(\lambda,\mathbf{m})\\
&\qquad+O^*(\lambda)N(\lambda)\Sigma^{-1}(\lambda)O(\lambda)
\end{aligned}\tag{13}$$

where $O(\lambda)=\Phi\mathbf{z}(\lambda)$. Substituting (13) into (12), we have (9) proven.

As a result, the posterior distribution of the vector $\mathbf{z}(\lambda)$ given the observed variable $\mathbf{x}$ can be approximated by

$$\begin{aligned}
P_\lambda(\mathbf{z}(\lambda)|\mathbf{x})&\propto P_\lambda(\mathbf{x}|\mathbf{z}(\lambda))P(\mathbf{z}(\lambda))\\
&=P_\lambda(\mathbf{x}|\mathbf{z}(\lambda))\aleph(\mathbf{z}(\lambda)|0,1)\\
&\propto\exp(\mathbf{z}^*(\lambda)\Phi^*\Sigma^{-1}(\lambda)\Xi(\lambda,\mathbf{m})-\frac{1}{2}\mathbf{z}^*(\lambda)\zeta^*(\lambda)\mathbf{z}(\lambda))\\
&\propto\exp(-\frac{1}{2}(\beta-\mathbf{z})^*\zeta(\lambda)(\beta-\mathbf{z}))
\end{aligned}\tag{14}$$

where $\beta=\zeta^{-1}(\lambda)\Phi^*\Sigma^{-1}(\lambda)\Xi(\lambda,\mathbf{m})$, and $\zeta(\lambda)=\mathrm{I}+\Phi^*\Sigma^{-1}(\lambda)N(\lambda)\Phi$, and I denotes identity matrix. This equation means: $\mathbf{E}\{\mathbf{z}(\lambda)\}=\beta$, and $\mathbf{E}\{\mathbf{z}^*(\lambda)\mathbf{z}(\lambda)\}=\zeta^{-1}(\lambda)$, where $\mathbf{E}$ denotes the expectation operator. Therefore, the expectation of $\mathbf{M}(\lambda)$ is given by

$$\begin{aligned}
&\mathbf{E}[\mathbf{M}(\lambda)]=\mathbf{m}+\Phi\mathbf{E}\{\mathbf{z}(\lambda)\}\\
&=\mathbf{m}+(\mathrm{I}+\Phi^2\Sigma^{-1}(\lambda)N(\lambda))^{-1}\Phi^2\Sigma^{-1}(\lambda)\Xi(\lambda,\mathbf{m})
\end{aligned}\tag{15}$$

We have

$$\hat{\mathbf{M}}(\lambda)=\mathbf{E}[\mathbf{M}(\lambda)]=\mathbf{m}+(\gamma+N(\lambda))^{-1}\Xi(\lambda,\mathbf{m})\tag{16}$$

Comparing with the conventional MAP, (16) shows that the relevance factor γ can be estimated by using Φ and $\Sigma(\lambda)$, i.e. $\gamma=\Phi^{-2}\Sigma(\lambda)$, or $\gamma_i=\Phi_i^{-2}\Sigma_i(\lambda), i=1,...,C$. As a matter of fact, we have to estimate Φ for the relevance factor in MAP algorithm.

3.2. Estimation of Φ for the relevance factor

According to Jensen's inequality, we have

$$\begin{aligned}
&\log\int\frac{P_\lambda(\mathbf{z},\mathbf{x})}{P_{\lambda_0}(\mathbf{z},\mathbf{x})}P_{\lambda_0}(\mathbf{z}|\mathbf{x})d\mathbf{z}\\
&=\log\int\frac{P_\lambda(\mathbf{z}|\mathbf{x})P_\lambda(\mathbf{x})}{P_{\lambda_0}(\mathbf{z}|\mathbf{x})P_{\lambda_0}(\mathbf{x})}P_{\lambda_0}(\mathbf{z}|\mathbf{x})d\mathbf{z}\\
&=\log\Big[\frac{P_\lambda(\mathbf{x})}{P_{\lambda_0}(\mathbf{x})}\Big]\int P_\lambda(\mathbf{z}|\mathbf{x})d\mathbf{z}\\
&=\log P_\lambda(\mathbf{x})-\log P_{\lambda_0}(\mathbf{x})\\
&\geq\int\Big(\log\frac{P_\lambda(\mathbf{z},\mathbf{x})}{P_{\lambda_0}(\mathbf{z},\mathbf{x})}\Big)P_{\lambda_0}(\mathbf{z}|\mathbf{x})d\mathbf{z}\\
&=\int\Big(\log\frac{P_\lambda(\mathbf{x}|\mathbf{z})P_\lambda(\mathbf{z})}{P_{\lambda_0}(\mathbf{x}|\mathbf{z})P_{\lambda_0}(\mathbf{z})}\Big)P_{\lambda_0}(\mathbf{z}|\mathbf{x})d\mathbf{z}\\
&=\int\big(\log P_\lambda(\mathbf{x}|\mathbf{z})\big)P_{\lambda_0}(\mathbf{z}|\mathbf{x})d\mathbf{z}\\
&\qquad-\int\big(\log P_{\lambda_0}(\mathbf{x}|\mathbf{z})\big)P_{\lambda_0}(\mathbf{z}|\mathbf{x})d\mathbf{z}\\
&=\Re_\lambda(\mathbf{x})-\Re_{\lambda_0}(\mathbf{x})
\end{aligned}\tag{17}$$

where $\Re_\lambda(\mathbf{x})=\int\big(\log P_\lambda(\mathbf{x}|\mathbf{z})\big)P_{\lambda_0}(\mathbf{z}|\mathbf{x})d\mathbf{z}=\mathbf{E}\Big[\log P_\lambda(\mathbf{x}|\mathbf{z}(\lambda))\Big]$. Therefore, the problem of maximizing $\log P_\lambda(\mathbf{x})$ could be replaced with maximization of $\Re_\lambda(\mathbf{x})$ with respect to λ. It can be derived as follows

$$\Re_\lambda(\mathbf{x})=\Theta(\lambda)+\mathbf{E}\big[\Omega(\lambda,\mathbf{z}(\lambda))\big]\tag{18}$$

We have the maximization solution for Φ by

$$\begin{aligned}
&\frac{\partial(\Re_\lambda(\mathbf{x}))}{\partial\Phi}\\
&=\mathbf{E}\big[\mathbf{z}^*(\lambda)\big]\Sigma^{-1}(\lambda)\Xi(\lambda,\mathbf{m})-\mathbf{E}\big[\mathbf{z}(\lambda)\mathbf{z}^*(\lambda)\big]\Phi N(\lambda)\Sigma^{-1}(\lambda)\\
&=\Sigma^{-1}(\lambda)\Big\{\mathbf{E}\big[\mathbf{z}^*(\lambda)\big]\Xi(\lambda,\mathbf{m})-\Phi\mathbf{E}\big[\mathbf{z}(\lambda)\mathbf{z}^*(\lambda)\big]N(\lambda)\Big\}\\
&=0
\end{aligned}\tag{19}$$

We have $\mathbf{E}\big[\mathbf{z}^*(\lambda)\big]\Xi(\lambda,\mathbf{m})=\Phi\mathbf{E}\big[\mathbf{z}(\lambda)\mathbf{z}^*(\lambda)\big]N(\lambda)$. Thus, we give the M-step for Φ as follows

$$\Phi=\mathbf{E}\big[\mathbf{z}^*(\lambda)\big]\Xi(\lambda,\mathbf{m})N^{-1}(\lambda)(\mathbf{E}\big[\mathbf{z}(\lambda)\mathbf{z}^*(\lambda)\big])^{-1}\tag{20}$$

Obviously, according to (14), the E-step is given by

$$\mathbf{E}\{\mathbf{z}(\lambda)\}=[\mathrm{I}+\Phi^*\Sigma^{-1}N(\lambda)\Phi]^{-1}\Phi^*\Sigma^{-1}(\lambda)\Xi(\lambda,\mathbf{m})\tag{21}$$

$$\mathbf{E}\{\mathbf{z}^*(\lambda)\mathbf{z}(\lambda)\}=[\mathrm{I}+\Phi^*\Sigma^{-1}(\lambda)N(\lambda)\Phi]^{-1}\tag{22}$$

Thus, Φ can be estimated by computing the expectation-maximization (EM) algorithm of (20 - 22) iteratively.

4. Performance evaluation

4.1. Evaluation on NIST LRE 2009

We conduct the experiments on NIST LRE 2009 core tasks. There are 23 target languages used for this evaluation, namely, Amharic, Bosnian, Cantonese, Creole (Haitian), Croatian, Dari, American English, Indian English, Farsi, French, Georgian, Hausa, Hindi, Korean, Mandarin, Pashto, Portuguese, Russian, Spanish, Turkish, Ukrainian, Urdu, and Vietnamese. Unlike LRE-2007 which has only telephony speech databases, LRE 2009 has two catagories of data sources named conversational

telephone speech (CTS) and Voice of America (VOA) narrowband speech. In this experiment, the CTS training database are collected from CallFriend, OHSU, LRE07 Train, OGI22 and SRE06; and the VOA data are mainly from the VOA3 database provided by NIST and LDC, VOA7 provided by NIST and VOA8 downloaded from the web-site. In the evaluation, 56-dimensional MFCC-SDC features with 7-1-3-7 delta-shift (refer to $\breve{N}$-$\breve{d}$-$\breve{P}$-$\breve{k}$ parameters in [9]) plus 7 static cepstral is computed after voice activity detection (VAD).

We evaluate the validity of the derived relevance factor by using a state-of-the-art language recognition system. We use the following Bhattacharyya kernel for SVM, which is called GMM-UBM Mean Interval (GUMI) system in [7],

$$K_{\text{Bhatt}}(\mathbf{X}_a, \mathbf{X}_b) = \sum_{i=1}^{C} \Big\{ \Big[\Big(\frac{\hat{\Sigma}_i(\lambda_\text{a}) + \Sigma_i}{2} \Big)^{-\frac{1}{2}} (\hat{\mathbf{M}}_i(\lambda_\text{a}) - \mathbf{m}_i) \Big]^T \times \Big[\Big(\frac{\hat{\Sigma}_i(\lambda_\text{b}) + \Sigma_i}{2} \Big)^{-\frac{1}{2}} (\hat{\mathbf{M}}_i(\lambda_\text{b}) - \mathbf{m}_i) \Big] \Big\} \quad (23)$$

In the GUMI language recognition system, we use 512 mixture components for GMM. We trained the diagonal matrix Φ by using EM algorithm with the initial Φ is $\Phi_i^{(0)} = (\Sigma_i)^{-\frac{1}{2}}$.

Outputs of individual classifiers are calibrated with separate linear backend followed by linear logistic regression (LLR). The calibrated scores are then combined via a final stage of LLR. Note that the development and evaluation data are grouped into 30, 10 and 3-second utterances. We only conduct the experiments on 30, 10 and 3-second utterances. Although the group of data are labeled under 30, 10 or 3-second categories, the actual duration of utterances varies. The training is done on the 30, 10 and 3-second development data respectively using the FoCal Multiclass toolkit [10]. Log-likelihood ratio score from all classifiers are stacked together and a linear backend is trained. This is then followed by an LLR stage. The scores are converted into log-likelihood ratio for final decision with a threshold set at zero. A development set was designed for the training of backend calibration. This development set consists of about 8000 trials for each category, and is built upon LRE07 augmented with additional trials taken from VOA3. The development set is split into two halves, one for training and the other for cross-validation.

In the performance evaluation, we consider three nominal durations: 30, 10 and 3 seconds in the testing. One is the GUMI system with data-dependent relevance factor (**GUMI-RF-DEP**), and another is with the relevance factor set to 16 (**GUMI-RF-FIX**). We show the results of the close-set tasks with nominal duration of 30 seconds, 10 seconds and 3 seconds. Table 1 reports the equal error rate (EER) and minimum detection cost function (min DCF) values respectively. It can be seen that the **GUMI-RF-DEP** system apparently outperforms **GUMI-RF-FIX**. With the data-dependent relevance factor, the relative improvement of 5.17% is obtained on the 30-second close-set task of LRE 2009.

Table 1: The comparison of the language recognition systems in terms of EER and minimum cost for LRE 2009 30s, 10s and 3s close-set tasks

LRE 2009, 30s	*EER*	*min. Cost × 100*
GUMI-RF-FIX	5.41 %	5.21
GUMI-RF-DEP	5.13 %	4.94
LRE 2009, 10s	*EER*	*min. Cost × 100*
GUMI-RF-FIX	11.02 %	10.80
GUMI-RF-DEP	10.65 %	10.37
LRE 2009, 3s	*EER*	*min. Cost × 100*
GUMI-RF-FIX	22.19 %	21.99
GUMI-RF-DEP	21.51 %	21.27

5. Summary

In GMM-SVM language recognition system, a GMM supervector is used to represent the language property of an utterance and serves as an input vector to the SVM. This requires the elimination of the negative effect of the database variation in order to manifest the saliency of the language characteristics. We studied the relevance factor of MAP for GMM, and derive the data-dependent relevance factor of MAP in supervector domain. We proposed the data-dependent relevance factor in the language recognition system. The efficacy of the data-dependent relevance factor is shown by using the Bhattacharyya-based GMM-supervector system on the LRE 2009 core tasks.

6. References

[1] M. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Trans. Speech Audio Process.*, vol. 4, no. 1, pp. 31-44, 1996.

[2] P. A. Torres-Carrasquillo, E. Singer, M. A. Kohler, R. J. Greene, D. A. Reynolds, and J. R. Deller Jr., "Approaches to language identification using Gaussian mixture models and shifted delta cepstral features," *Int. Conf. on Spoken Lang. Process.*, 2002, pp. 89C92.

[3] J. L. Gauvain and C-H. Lee, "Maximum *a posteriori* estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Trans. Speech Audio Process.*, vol. 2, pp. 291-298, 1994.

[4] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digit. Signal Process.*, vol. 10, pp. 19-41, 2000.

[5] P. Kenny, "Joint Factor Analysis of Speaker and Session Variability : Theory and Algorithms," CRIM, Montreal, *Technical Report, CRIM-06/08-13*, 2005.

[6] C. H. You, H. Li, and K. A. Lee, "A GMM-supervector approach to language recognition with adaptive relevance factor," *18th Europ. Signal Process. Conf.*, EUSIPCO, pp. 1993-1997, Aalborg, Aug. 2010.

[7] C. H. You, K. A. Lee, and H. Li, "An SVM kernel with GMM-supervector based on the Bhattacharyya distance for speaker recognition," *IEEE Signal Processing Letters*, vol. 16, no. 1, pp. 49-52, Jan. 2009.

[8] http://www.itl.nist.gov/iad/mig/tests/lre/2009/

[9] W. M. Campbell, E. Singer, P. A. Torres-Carrasquillo, and D. A. Reynolds, "Language Recognition with Support Vector Machines," *Proc. Odyssey: The Speaker and Lang. Recog. Workshop* Toledo, pp. 41-44, 2004.

[10] N. Brümmer, "FoCal Multi-class: Toolkit for Evaluation, Fusion and Calibration of Multi-class Recognition Scores," Available: http://niko.brummer.googlepages.com/focalmulticlass.