Acoustic Forest for SMAP-based Speaker Verification

Sangeeta Biswas[1]*, Marc Ferras, Koichi Shinoda, Sadaoki Furui*

Department of Computer Science, Tokyo Institute of Technology, Tokyo, Japan

[1]sangeeta@ks.cs.titech.ac.jp

Abstract

In speaker verification, structural maximum-a-posteriori (SMAP) adaptation for Gaussian mixture model (GMM) has been proven effective, especially when the speech segment is very short. In SMAP adaptation, an acoustic tree of Gaussian components is constructed to represent the hierarchical acoustic space. Until now, however, there has been no clear way to automatically find the optimal tree structure for a given speaker. In this paper, we propose using an *acoustic forest*, which is a set of trees, for SMAP adaptation, instead of a single tree. In this approach, we combine the results of SMAP adaptation systems with different acoustic trees. A key issue is how to combine the trees. We explore three score fusion techniques, and evaluate our approach in the text-independent speaker verification task of the NIST 2006 SRE plan using 10-second speech segments. Our proposed method decreased EER by 3.2% from the relevant MAP adaptation and by 1.6% from the conventional SMAP with a single tree.

Index Terms: speaker verification, text-independent, short speech, MAP, SMAP

1. Introduction

Recent research on text-independent speaker verification has mainly focused on the problem of channel compensation of GMM-based speaker models. However, when a speech segment is very short (e.g. 10 seconds), extra attention goes to model adaptation for making a speaker-dependent GMM from the speaker-independent GMM called universal background model (UBM). For 10-second speech segments, Vogt et al. [1] proposed using speaker subspace MAP adaptation for factor analysis (FA) modeling. Fauve et al. [2] proposed a well-tuned speech detection front-end for improving frame selection in eigenvoice modeling. Kenny et al. [3] extended joint factor analysis (JFA) to model within-session-variability over a shorter time span. The adaptation part of all these methods did not use structural modeling for sharing parameters in the acoustic space.

We try to handle such short speech segments by structural modeling using the structural maximum-a-posteriori (SMAP) adaptation technique. This technique was first proposed by Shinoda et al. [4] for speech recognition. In speaker verification, Liu et al. [5] and Xiang et al. [6] successfully applied it to speech segments of 2 minutes or shorter. In SMAP adaptation, a single tree structure has been used to model the acoustic space of all the speakers. That is, we have implicitly assumed that the hierarchical structure of the acoustic space can be shared among all the speakers. During our work on speaker verification, however, we notice that a single tree structure is not always optimal for modeling the acoustic space of every speaker. Ideally, different tree structures should be provided for different speakers. However, until now no methods for obtaining such trees are known. In this paper, we propose using an acoustic forest, which is a set of trees instead of a single tree, in order to solve this problem.

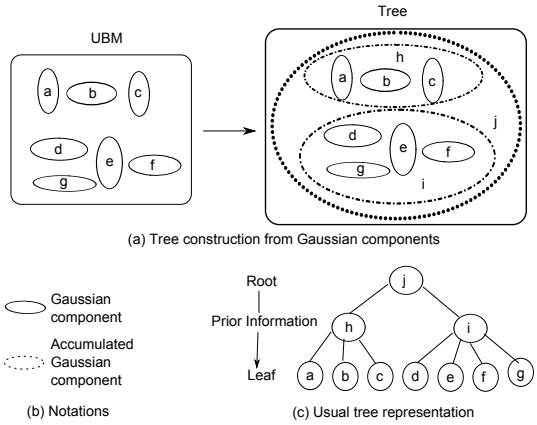

Figure 1: *An example of a tree structure of Gaussian components in SMAP. Each of a, b, ..., g is a Gaussian component of a UBM. h, i and j are parent Gaussians of* $\{a, b, c\}$*,* $\{d, e, f, g\}$ *and* $\{h, i\}$*, respectively.*

The remainder of the paper is organized as follows. A brief description of SMAP adaptation is given in Section 2. Section 3 illustrates acoustic forest. In Section 4, we describe our GMM-SVM based system. In Section 5 and Section 6, we describe our experimental setup and results, respectively. Section 7 gives some conclusions.

2. SMAP adaptation

The SMAP adaptation was proposed to keep the desirable asymptotic properties of relevance MAP while dealing with the problem of data sparseness by using a tree structure. The SMAP-based method have two steps. In the first step, a tree is obtained by clustering Gaussian components of the UBM as shown in Fig. 1(a). The root node of the tree represents the whole acoustic space and each of the non-leaf nodes has a Gaussian component that summarizes its child node distributions. Each leaf node corresponds to a Gaussian component in the UBM as shown in Fig. 1(c). In the second step, a speaker-dependent model is obtained by using the distribution of each non-leaf node as the prior for parameters of its child nodes. These two steps are briefly described in the following two subsections.

2.1. Tree construction

In order to define a tree structure, we first provide the number of layers L and the number of branches $B_r^{(l)}$ from a node r at the l-th layer prior to clustering[1]. For clustering, we use the symmetric Kullback-Leibler (KL) divergence as the distance measure between two Gaussian components. Assuming the covariance matrices to be diagonal, the KL divergence between two Gaussian components, $g_a(\cdot)$ and $g_b(\cdot)$, can be written as

$$d(a,b) = \sum_{i=1}^{F}[\frac{\sigma_a^2(i) - \sigma_b^2(i) + (\mu_b(i) - \mu_a(i))^2}{\sigma_b^2(i)} + \frac{\sigma_b^2(i) - \sigma_a^2(i) + (\mu_a(i) - \mu_b(i))^2}{\sigma_a^2(i)}], \quad (1)$$

where $\mu_a(i)$ is the i-th element of F-dimensional mean vector μ_a and $\sigma_a^2(i)$ is the i-th diagonal element of covariance matrix $\sum_a$.

The algorithm for obtaining a tree from a UBM with M Gaussians is given below:

1. Set:
   (a) k to be the root node
   (b) G_k to be a set of all the M Gaussians governed by node k,
   (c) $B_k^{(1)}$ to be the number of children of node k
   (d) l to be 1.
2. Calculate the node pdf g_k for node k using the following formulas:

$$\mu_k(i) = \frac{1}{M_k}\sum_{m \in G_k} \mu_m(i), \quad (2)$$

$$\sigma_k^2(i) = \frac{1}{M_k}[\sum_{m \in G_k}(\sigma_m^2(i) + \mu_m^2(i)) - M_k\mu_k^2(i)], \quad (3)$$

   where M_k is the number of Gaussian components included in G_k.
3. If l is equal to L, stop clustering, else go to Step 4.
4. Compute the initial pdf for n child nodes using the $minimax$ method:
   (a) Find n Gaussian components from G_k:
      i. The 1st Gaussian is $g_{c_1}(\cdot) = g_{\hat{m}}(\cdot)$ where

$$\hat{m} = \arg\max_m d(m,k). \quad (4)$$

      ii. The remaining $(n-1)$ Gaussians will be $g_{c_p}(\cdot) = g_{\hat{m}}(\cdot)$ where

$$\hat{m} = \arg\max_m \min_{q \in G_{ck}} d(m, c_q). \quad (5)$$

      Here G_{ck} is the set of Gaussians already assigned to the child nodes of node k, $1 \le p \le n-1$ and $1 \le q \le n-2$.
   (b) Interpolate the node pdf of node k and the initial node pdf of each child node c_p to create a new node pdf for c_p as follows:

$$\hat{\mu}_{c_p}(i) = (1-\alpha)\mu_k(i) + \alpha\mu_{c_p}(i), \quad (6)$$

$$\hat{\sigma}_{c_p}^2(i) = (1-\alpha)(\sigma_k^2(i) + \mu_k^2(i)) + \alpha(\sigma_{c_p}^2(i) + \mu_{c_p}^2(i)) - \hat{\mu}_{c_p}, \quad (7)$$

      where $0 \le \alpha \le 1$.
5. Repeat the following k-means procedures until the grand sum of distances, $\mathcal{GD}$, converges:
   (a) For each Gaussian component in G_k, calculate the distance from it to each child node pdf of the l-th layer by using Eq. (1), and assign it to the nearest child node.
   (b) Recalculate the child node pdf by using Eq. (2) and Eq. (3).
   (c) Using Eq. (1), calculate the sum of distances, $\mathcal{D}$, from each child node to each of its mixture components and then obtain $\mathcal{GD}$ by accumulating all $\mathcal{D}$.
6. Set each child node to be node k and its corresponding subset of Gaussian components to be G_k. Increase l and go to Step 4.

2.2. Adaptation

The formulation of SMAP adaptation is similar to that of the relevance MAP [7], except that it uses hierarchical priors and normalized pdfs in the formulation. The adaptation steps for each node p using adaptation data $X = \{x_1, x_2, ..., x_T\}$ are:

1. Transform each sample vector x_t into a vector y_{mt} for each mixture component m as follows:

$$y_{mt}^{(p)} = \mathbf{\Sigma}_m^{-1/2}(x_t - \mu_m^{(p)}), \quad (8)$$

   where $t = 1, 2, ..., T$ and $m = 1, 2, ..., M^{(p)}$.
2. Estimate the normalized pdf $\mathcal{N}(Y^{(p)}|\nu, \eta)$ for $Y_m^{(p)} = \{y_{m1}^{(p)}, y_{m2}^{(p)}, ..., y_{mT}^{(p)}\}$, where $\nu^{(p)}$ and $\eta^{(p)}$ represent the shift and rotation needed to compensate for the distortion, i.e., to adapt the model parameters to the data. When there is no mismatch between the training and adaptation data, then $\nu^{(p)} = \vec{0}$ and $\eta^{(p)} = I$. The ML estimation of the mean vector of the normalized pdf is calculated as follows:

$$\tilde{\nu}^{(p)} = \frac{\sum_{t=1}^{T}\sum_{m=1}^{M^{(p)}}\gamma_{mt}^{(p)}y_{mt}^{(p)}}{\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{mt}^{(p)}}, \quad (9)$$

   where $\gamma_{mt}^{(p)}$ is the occupation probability for Gaussian m at tree node p and time t.
3. Calculate the hierarchical prior

$$\hat{\nu}^{(p)} = \frac{N^{(p)}\tilde{\nu}^{(p)} + \tau\hat{\nu}^{(p-1)}}{N^{(p)} + \tau}, \quad (10)$$

   where $N^{(p)} = \sum_{t=1}^{T}\sum_{m=1}^{M^{(p)}}\gamma(p)_{mt}$ is the average number of frames assigned to node pdf p and τ is the MAP relevance factor that weights the priors at the parent node $p-1$.

[1] At present, we have no automatic ways to obtain the optimal numbers for branches and layers for each speaker

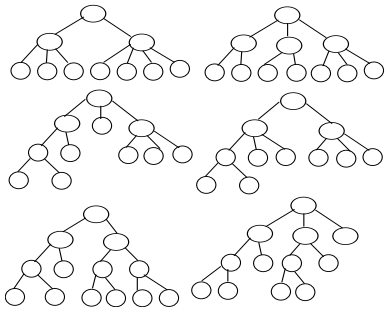

Figure 2: *Acoustic Forest.*

4. Compute the SMAP estimate of the mean vector

$$\hat{\mu}_m^{(p)} = \mu_m^{(p)} + \mathbf{\Sigma}_m^{1/2}\hat{\nu}^{(p)}, \tag{11}$$

where $\mu_m^{(p)}$ is the unadapted mean vector for Gaussian m of node p.

When no sufficient amount of adaptation data is available for a Gaussian component, it is not shifted in relevance MAP. In SMAP adaptation, on the other hand, it takes prior information from its parent Gaussian, and accordingly, every Gaussian component is shifted from its position in UBM. Figure 1(a) shows a schematic example, where a, b, c get prior information from h, d, ..., g from i, and $\{h, i\}$ from j.

3. Acoustic forest

In SMAP adaptation, a tree structure obtained by clustering Gaussians offers a convenient way to capture the hierarchical structure of the acoustic space of the human voice. Different speakers have different acoustic spaces depending on factors such as their language, accents or pronunciation particularities. It is therefore reasonable to think that the optimal tree structure differs from speaker to speaker. In other words, some tree structures may be adapted more efficiently to some speakers than others. However, to find the optimal tree structure for every speaker is computationally expensive when the number of speakers is large, and demands a large amount of data. The easiest solution of this problem is to use a set of trees, assuming that the verification accuracy will generally be higher when combining the decisions of the set of trees, rather than using only a single tree. We define the set of trees as an acoustic forest. In the acoustic forest, the number of layers and number of brances of each node will vary from tree to tree. Figure 2 shows the acoustic forest having five tree structures for the UBM mentioned in Fig. 1(a).

There are different ways to combine the decisions of multiple SMAP adapted systems with different tree structures. Like the random forest algorithm [8], we can use a voting approach or we can fuse the scores of different systems and take the decision by setting a threshold on the fused score. In this paper we use score fusion techniques.

3.1. Score fusion

Let $s_1, s_2, ..., s_L$ be the L scores of L SMAP adapted systems. Then the fused score, $\hat{S}$ of the claimed speaker can be calculated in the following ways:

- Maximization

$$\hat{S} = \max(s_1, s_2, ..., s_L) \tag{12}$$

- Sum

$$\hat{S} = \sum_{l=1}^{L} s_l \tag{13}$$

- Multilayer Perceptron (MLP)

$$\hat{S} = \frac{1}{1 + \exp(-(\sum_{h=1}^{H} w_{h,o} y_h + \delta_{h,o}))}, \tag{14}$$

where H is the number of neurons in the hidden layer, $w_{h,o}$ and $\delta_{h,o}$ are the weights connecting the hidden layer and the single output neuron of an MLP. y_h is the output of the h-th neuron of the hidden layer which is obtained by the following sigmoid function:

$$y_h = \frac{1}{1 + \exp(-(\sum_{l=1}^{L} w_{l,h} s_l + \delta_{l,h}))} \tag{15}$$

where $w_{l,h}$ and $\delta_{l,h}$ are the weights connecting the l-th neuron of the input layer and the h-th neuron of the hidden layer, and s_l is the score of the l-th system.

4. GMM-SVM system

In our evaluation, we use a GMM-SVM system proposed by Campbel et al. [9]. First we train a speaker-independent GMM using hours of speech by hundreds of speakers. After training the UBM, adaptation methods are used to make a speaker-dependent GMM from UBM using approximately 10 seconds of speech data for the target speaker. After making the GMM, a supervector is made by stacking the mean vectors of the GMM of the target speaker and a set of background speakers, used as negative data in the SVM classifier. Then the supervectors are used as inputs to a SVM with a linear kernel to train a GMM-SVM system for the target speaker. The score for each test speech segment X is calculated as follows:

$$\mathcal{S}_X = w\mathcal{M}_X + b, \tag{16}$$

where b is a constant, $\mathcal{M}_X$ is the supervector, and w is calculated as follows:

$$w = \sum_{i=1}^{R} \beta_i c_i \hat{\mathcal{M}}_i, \tag{17}$$

where R is the number of supportvectors, $\hat{\mathcal{M}}_i$ is the i-th supportvector, c_i is the class ID {1,-1} of $\hat{\mathcal{M}}_i$, β_i is the Lagrange multiplier, $\beta_i > 0$, and $\sum_{i=1}^{R} \beta_i c_i = 0$.

5. Experimental setup

Performance of our speaker verification system was measured by carrying out experiments on the 10sec4w-10sec4w task of the 2006 NIST SRE [10]. In this task, the length of each training and test segment is approximately 10 seconds. There are 2971 true trials and 30584 false trials for 731 speakers among which 316 are males and 415 are females. We trained one gender-independent UBM and two gender-dependent UBMs using 4806 speech segments from the NIST SRE 2004 training database. Each speech segment was 2.5 minutes long on average. Among 4806 speech segments, 242 speech segments of male speakers and 362 speech segments of female speakers were selected as speech segments of background speakers. As a development dataset and for T-Normalization, we used the NIST SRE 2005 training database.

Regarding feature extraction, we first removed the non-speech part from the speech segments using the information in

the transcript files. We broke each segment into frames of 30 ms, with a frame rate of 100 frames/sec. We pre-emphasized each frame with a pre-emphasis factor of 0.97 and applied a Hamming window. We computed 15 perceptual linear prediction (PLP) coefficients and mel-frequencey cepstral coefficients (MFCCs), augmented with energy, first and second-order derivatives, resulting in 48 features per frame. Cepstral mean subtraction was applied to remove static channel effects.

The performance measure was equal error rate (EER) and minimum detection cost (MDC). We chose ten different tree structures heuristically for the acoustic forest. A three layer MLP with three hidden neurons in the hidden layer was trained for score fusion. The feature extraction and GMM part were implemented by using the hidden markov model toolkit (HTK). The SVM classifier was made by LIBSVM, and MLP for score fusion was implemented by MATLAB.

6. Results

First we conducted an experiment on relevance MAP-adapted GMMs with 32 Gaussian components. By setting the relevance factor equal to 10, we found that the system using the gender-dependent UBM was better than the system using the gender-independent UBM, and PLP outperformed MFCC. We also noticed that the performance of our MAP adapted system improved when we increased the number of Gaussian components until 512, decreased the relevance factor to 1, and did not use the delta-delta coefficients.

Table 1 shows the EER(%) and MDC of our MAP and SMAP adapted system where we used the gender-dependent UBM with 512 Gaussian Components, 32 dimensional PLP feature vector (i.e. 15 PLP + 15 ΔPLP + E + ΔE), and set the relevance factor to 1. Most of the SMAP-adapted systems outperform the relevance MAP-adapted system. Error rates of SMAP-adapted systems consistently decrease as the number of nodes gets larger. The best relative improvement for individual SMAP systems, around 3.2%, is obtained for the 21_21 tree structure-based system without T-Normalization. T-Normalization helped to drop the EER slightly for both MAP and SMAP adapted systems.

As shown in Table 2, score fusion techniques improved the SMAP adapted system by decreasing only EER. Two fusion techniques, sum and MLP, gave the same performance. 1.6% relative improvement was gained in EER compared to the single tree structure-based system.

7. Conclusions

We have proposed to grow an acoustic forest with different tree structures for SMAP adapted text-independent speaker verification. We have implemented three types of score fusion techniques in order to combine the decision of several SMAP adapted GMM-SVM systems. By doing experiment on the 10sec4w-10sec4w task of NIST 2006 SRE we showed that score fusion techniques helped to improve the perforformance of SMAP adapted system. Two fusion techniques, sum and MLP, gave an improvement in EER. In this paper we only gave a comparative figure of relevance MAP and SMAP adapted systems for short speech segments. In future work we plan to compare SMAP adaptation with other adaptation techniques, such as eigenvoice modeling.

Table 1: *MDC and EER for GMM-SVM systems using MAP and SMAP adaptation on the 10sec4w-10sec4w task of 2006 NIST SRE. The design of a tree is written as n_1_n_2 where n_l represents the maximum number of child nodes belonging to each node of the l-th layer. Each leaf node corresponds one component in GMM.*

Fusion	No Norm		T-Norm	
	EER(%)	MDC	EER(%)	MDC
MAP	**27.7**	**0.0917**	**27.4**	**0.0910**
SMAP 3_3	28.2	0.0959	27.8	0.0941
SMAP 5_5	27.9	0.0943	27.6	0.0921
SMAP 7_7	27.7	0.0943	26.9	0.0917
SMAP 9_9	27.4	0.0937	26.9	0.0918
SMAP 11_11	26.9	0.0936	**26.6**	0.0918
SMAP 13_13	27.1	0.0932	26.6	0.0913
SMAP 15_15	27.3	0.0930	26.9	**0.0909**
SMAP 17_17	27.2	0.0933	27.0	0.0910
SMAP 19_19	27.2	0.0927	26.9	0.0913
SMAP 21_21	**26.8**	**0.0922**	26.9	0.0915

Table 2: *Comparison of the EER and the MDC for fusion of 15 SMAP adapted systems with and without T-Norm on the NIST 2006 SRE 10sec4w-10sec4w task.*

Fusion	No Norm		T-Norm	
	EER(%)	MDC	EER(%)	MDC
Maximization	27.2	0.0945	26.5	0.0915
Sum	26.5	0.0928	26.2	0.0909
MLP	26.5	0.0922	26.2	0.0908

8. References

[1] R. Vogt, C. Lustri, and S. Sridharan, "Factor analysis modelling for speaker verification with short utterances," in *Proc. IEEE Odyssey:The Speaker and Language Recognition Workshop*, Jan. 2008.

[2] B. Fauve, N. P. N. Evans, J. F. Bonastre, and J. Mason, "Influence of task duration in text-independent speaker verication," in *Proc. Interspeech*, Aug. 2007, pp. 794–797.

[3] P. Kenny and N. Dehak, "Factor analysis conditionning," in *Report from JHU workshop*, 2008, pp. 20–42.

[4] K. Shinoda and C.-H. Lee, "A structural bayes approach to speaker adaptation," *IEEE Trans. on Speech and Audio Processing*, vol. 9, no. 3, pp. 276–287, Mar. 2001.

[5] M. Liu, E. Chang, and B.-Q. Dai, "Hierarchical gaussian mixture model for speaker verification," in *Proc. ICSLP*, Sep. 2002.

[6] B. Xiang and T. Berger, "Efficient text-independent speaker verification with structural gaussian mixture models and neural network," *IEEE Trans. on Speech and Audio Processing*, vol. 11, pp. 447–456, Sep. 2003.

[7] J.-L. Gauvain and C.-H. Lee, "Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains," *IEEE Trans. on Speech and Audio Processing*, vol. 2, pp. 291–298, 1994.

[8] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

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

[10] "http://www.itl.nist.gov/iad/mig/tests/sre/2006/sre-06_evalplan-v9.pdf."