COMBINING EIGENVOICES AND STRUCTURAL *MLLR* FOR SPEAKER ADAPTATION

Fabrice Lauri, Irina Illina, Dominique Fohr

Speech Group, LORIA - INRIA
B.P. 239 - 54506 Vandœuvre-lès-Nancy - FRANCE
{lauri,illina,fohr}@loria.fr

ABSTRACT

This paper considers the problem of speaker adaptation of acoustic models in speech recognition. We have investigated four different possible methods which integrate the concepts of both Structural Maximum Likelihood Linear Regression (*SMLLR*) and EigenVoices technique to adapt the Gaussian means of the speaker independant models for a new speaker. The experiments were evaluated using the speech recognition engine ESPERE on the data of the corpus *Resource Management*. They show that all of the proposed methods can improve the performances of an ASRS in supervised batch adaptation as efficiently as *SMLLR* and EigenVoices-based techniques whatever the amount of adaptation data is available. For an unsupervised incremental adaptation, only the approaches *SMLLR→EV* and *SMLLR→SEV* seemed to give the best results.

1. INTRODUCTION

Reducing acoustic mismatches due to speaker variability between the training conditions and the testing conditions is a major problem in automatic speech recognition systems (ASRS). This problem is particularly difficult for rapid adaptation, when the available amount of adaptation data is small. Among the speaker adaptation techniques which tackle this problem, *MLLR* [1], [2] and EigenVoices [3], [4], [5] have shown to rapidly improve the performances of an ASRS.
Whereas *MLLR* becomes efficient only after a certain number of adaptation utterances have been pronounced, EigenVoices can improve the performances of an ASRS even if only one adaptation utterance has been used (see Fig. 1). This outstanding result of EigenVoices can be explained by the fact that unlike *MLLR*, it employs *a priori* information about the inter-speaker variations, which enable it to estimate much less parameters than *MLLR*. Still, the performances of EigenVoices technique quickly saturate as more adaptation data becomes available.

From these establishments, several speaker adaptation techniques recently try to integrate the advantages of *MLLR* and EigenVoices scheme to rapidly adapt to a new speaker the Gaussian means of the speaker-independant (SI) models.
The scheme presented in [6] extends the standard EigenVoices technique to large-vocabulary continuous speech recognition by training the acoustic models of each training speaker from SI models with the help of *MLLR* and *MAP*. In [7], the eigenspace representing the inter-speaker variations is build using *Principal Component Analysis (PCA)* from the parameters of the *MLLR* regression matrices obtained for each training speaker. The regression matrices computed for the adapted models of the new speaker are then constrained to be located in the space spanned by the first eigen-matrices. This method thus solves the problem of huge memory requirements of the EigenVoices technique, for the number of parameters of the regression matrices is much smaller than the parameters of a speaker-independant system. In [8], the authors propose three approaches which combine *MLLR* and EigenVoices adaptation. The Approach B exposed in [8] gives similar results to EigenVoices technique but requires far less online memory and computation load. In this approach, a new fast algorithm for maximum-likelihood coefficient estimation is used and the selection of the eigenspace includes SI-model information.
All of the above approaches have shown that the integration of *MLLR* and EigenVoices adaptation is fairly robust and reliable. Nevertheless, all of them are performed in batch mode with only a small amount of adaptation data. Real applications require that adaptation techniques are able to improve the performance of an ASRS continuously, as more utterances are pronounced by a new speaker.

This paper focuses to know how the performances obtained with techniques combining EigenVoices and *SMLLR* evolve as more adaptation data become available. We have made some investigations on different possible methods which integrate the concepts of both *SMLLR* and EigenVoices for speaker adaptation in supervised batch mode and in unsupervised incremental mode.
The remainder of this paper is organized as follows. Structural *MLLR* algorithm is introduced in Section 2. Section 3 reviews the regular version of EigenVoices algorithm and proposes a structural version of it. Section 4 presents four different methods which combine either EigenVoices and *SMLLR* techniques, or Structural EigenVoices and *SMLLR* techniques. The proposed methods are referred to as Approaches *EV→SMLLR, SEV→SMLLR, SMLLR→EV* and *SMLLR→SEV*. Section 5 evaluates the different proposed methods using data from the *Resource Management (RM)* corpus. Finally, concluding remarks and future research issues are given in Section 6.

2. STRUCTURAL *MLLR*

The Structural version of *MLLR* [2] is able to adjust the number of linear regression matrices $\{W_1, W_2, \cdots, W_N\}$ that will be applied to the Gaussian mean vectors according to the available amount of adaptation data. This flexibility is realized by using a binary tree structure that cluster the gaussians of the SI-models. Each tree node G_i with a transformation matrix W_i is called a regression class. Let $\gamma_t(g_{(i,m)})$ be the occupation probability of gaussian m of the regression class i at time t, $S_i =$

$\sum_{m=1}^{M_i} \sum_{t=1}^{T} \gamma_t(g_{(i,m)})$ be the number of observations associated to the set of M_i gaussians belonging to the regression class i. For each leaf node which possesses more than θ_{SMLLR} observations, that is $S_i \geq \theta_{SMLLR}$, the associated matrix is estimated by using the set of gaussians of the node. For the leaf nodes that have not enough observations, the associated matrix is estimated by using the gaussians of the closest father node which has enough observations. This regression classes generation process can update the parameters of the gaussians by using robust estimated transformation matrices.
The adapted Gaussian mean vector $\hat{\mu}_{(i,m)}$ of the gaussian m of the class i is then obtained by the transformation $\hat{\mu}_{(i,m)} = W_i \ \xi_{(i,m)}$ where $\xi_{(i,m)}$ is the extended vector of the Gaussian mean $\mu_{(i,m)}$ such as $\xi_{(i,m)} = [1 \ \mu^{'}_{(i,m)}]^{'}$.

3. EIGENVOICES

EigenVoices technique uses *a priori* information about inter-speaker variations to constrain the adapted models to be located in a dimensionality reduced speaker-space. The speaker space reduced in dimension is obtained by applying a dimensionality reduction technique [1] to a set of T supervectors of dimension D extracted from T well-trained speaker-dependant (SD) models. A supervector is made up with the parameters that have to be adapted. Typically, it consists in the concatenation of all of the Gaussian mean vectors of all of the models of a speaker-dependant system if only Gaussian means need to be adapted.
This offline step yields T supervectors of dimension D, called the eigenvectors. To get the reduced speaker-space, only the K first eigenvectors $\{e_1, e_2, \cdots, e_K\}$ with $K < T << D$ are kept. Related to an origin e_0 [2], these K eigenvoices, which capture most of the variation of the training data, span the reduced speaker-space of dimension K.

3.1. Regular version

In the regular version of EigenVoices technique (EV), a new speaker is located in the reduced speaker-space by a vector of $K+1$ weights $\{w_0, w_1, \cdots, w_K\}$.
All of the Gaussian mean vectors $\hat{\mu}_i$ of the adapted models are then updated using the equation $\hat{\mu}_i = \sum_{k=0}^{K} w_k \ e_k$ with $i = 1, 2, \cdots, N$, where N is the total number of gaussians of the speaker-adapted system.
The $K+1$ weights are generally estimated using *Maximum Likelihood Eigen-Decomposition (MLED)* [9] to maximize the likelihood of the adaptation data. The other HMM parameters are obtained from the corresponding SI-model parameters.

3.2. Structural version

The structural version of EigenVoices (SEV) borrows the flexibility of *SMLLR* by also using a Gaussian binary tree structure to adjust the adaptation parameters with the available amount of adaptation data. Structural EigenVoices thus avoid the early saturation encountered by its regular counterparts when more adaptation data is available.
A regression class i in Structural EigenVoices represents a tree node G_i with a set of $K+1$ weights that will be applied only to the corresponding gaussians belonging to the node G_i. The regression classes generation process in Structural EigenVoices is the same that the one in *SMLLR* : the $K+1$ weights are estimated only if more than θ_{SEV} observations have been gathered in the class i.

As the number of adaptation parameters which need to be estimated in a regression class in Structural EigenVoices is smaller than in *SMLLR*, the value of θ_{SEV} will be lower than the value θ_{SMLLR}. For this reason, theoretically, SEV is able to adapt independently the Gaussian means of more regression classes than *SMLLR*. Nevertheless, to avoid poor estimates of the adaptation parameters due to a bad value of θ_{SEV}, we assume that the regression classes generation process in SEV is triggered only if the total number of observations is greater than some predetermined threshold α_{SEV}.

4. COMBINING *SMLLR* WITH EIGENVOICES

We propose hereafter four possible methods which integrate the concepts of *SMLLR* and EigenVoices technique and which can be easily applied for speaker adaptation in both supervised batch mode and unsupervised incremental mode.

4.1. Approaches *EV→SMLLR* and *SEV→SMLLR*

These approaches consist in first obtaining adapted models with the help of either EigenVoices technique (*EV→SMLLR*) or Structural EigenVoices (*SEV→SMLLR*). The adapted models obtained at the previous step are then used as initial models by *SMLLR* to provide the final adapted models.
These approaches suggest that *SMLLR* adaptation is more efficient after an adaptation with one of the EigenVoices-based (EV or SEV) techniques.

4.2. Approaches *SMLLR→EV* and *SMLLR→SEV*

These approaches swap the two steps involved in the two previous techniques. Hence, they consist in first obtaining adapted models with the help of *SMLLR*. A supervector e_{K+1} is then extracted from the adapted models generated at the previous step and another weight w_{K+1} is estimated with the help of EigenVoices technique (*SMLLR→EV*) or Structural EigenVoices (*SMLLR→SEV*) to provide the final adapted models.
Here we assume that the EigenVoices adaptation is more robust after an adaptation with *SMLLR*.

5. EXPERIMENTAL EVALUATION

5.1. Database and System

EigenVoices, *SMLLR* and the proposed approaches have been implemented into the automatic speech recognition system ESPERE [3] [10] and evaluated on the corpus *Resource Management (RM)*.
The speech signals in *RM* are sampled at 16 kHz and were parameterized into the 11 MFCCs $C1$ to $C11$ and the 12 first and second order time derivatives of $C0$ to $C11$, yielding a 35-dimensional feature vector.
The speaker-independant training set of RM1 was used to train

[1] Principal Component Analysis (PCA) for instance

[2] e_0 can be the average supervector of all the SD models or the supervector extracted from the SI models.

[3] ESPERE is a first order HMM-based speech recognition toolbox developed at LORIA.

the acoustic models of both the speaker-independant system and the speaker-dependant systems. This set groups together 25 female and 55 male american native speakers. Each speaker pronounced 40 training utterances, for a total of 3200 utterances. The acoustic models of the speaker-independant system were trained by performing 20 iterations of the Baum-Welch algorithm ; each speaker-dependant system was trained by adapting the speaker-independant system using 10 iterations of *Structural Maximum A Posteriori (SMAP)* [11]. We used the speech data from 2 female and 2 male speakers of the speaker-dependant set RM2 for the adaptation phase and the recognition phase. Each speaker uttered 600 training sentences (used for the adaptation phase only) and 120 sentences (used for the recognition phase).

The acoustic units in the speaker-independant system and in each speaker-dependant system are represented by 45 HMMs with 3 states and a HMM with one state to handle silence and short pause. The probability density function of each state is modelled by a mixture of 8 gaussians. Speech recognition experiments were conducted by using the regular *word-pair* grammar of *RM*.

The *LBG* method combined with the *K-Means* procedure [4] were used to build the gaussian tree handled by *SMLLR* and by the Structural version of EigenVoices.
The accumulation of the sufficient statistics during the incremental process was carried out using the procedure proposed in [12]. This procedure consists in computing for a given gaussian g its sufficient statistics by adding to the sufficient statistics computed for the current $n-th$ utterance using the previous adapted system the sufficient statistics gathered before the $n-th$ utterance have been pronounced. These sufficient statistics are then used to estimate the adaptation parameters.
For each experiments, the binary tree used by *SMLLR* and Structural EigenVoices was built from the SI models. Its depth was set to 6. The value of the threshold θ_{SMLLR} used to robustly estimate the adaptation parameters was set to 1000 ; the value of the threshold θ_{SEV} was set to 60, the value of α_{SEV} was set to 1000. This parameterization seemed to provide the best results.

5.2. Experimental Results

ESPERE engine was evaluated in speaker-dependant mode, in speaker-independant mode and in speaker-adapted mode. All the subsequent results represents the average word accuracy (*WA*) of four speakers, by taking a confidence interval of $\pm 1\%$, with a risk of 5%. The *WA* of the speaker-dependant system is of 94.9%; the *WA* of the speaker-independant system is of 88.6%.

The figure 1 shows the results of the regular version of EigenVoices, Structural EigenVoices and *SMLLR*. EV is better than *SMLLR* for the first two utterances. EigenVoices need to estimate less parameters than *SMLLR* with the same amount of adaptation data, which can be done robustly in the case of EigenVoices adaptation. From the third utterance, *SMLLR* gives better results than EigenVoices adaptation, which starts to saturate at this point. This is due to the limited number of adaptation parameters which are unable to capture all of the information gathered in the adaptation data. Structural EigenVoices gives results slightly better than *SMLLR* from the third utterance to the 24th utterance, for SEV is able to define more regression classes and thus to transform independently more Gaussian means than *SMLLR*. SEV then starts to saturate from the 25th utterance, certainly for the same reason that standard EigenVoices adaptation saturates at the third utterance. Hence, the structural version of EigenVoices push back the point at which the performance saturation begins.

[4]The *Mahalanobis* distance was used as the distance measure between a gravity center of a node and a gaussian.

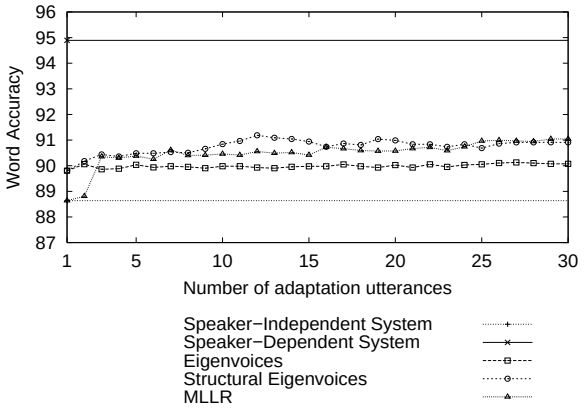

Fig. 1. Comparison between the regular version of EigenVoices (EV), Structural EigenVoices (SEV) and *SMLLR* in unsupervised incremental adaptation

The table 1 presents the results of the four proposed methods compared with *SMLLR* and the EigenVoices-based techniques for a supervised batch adaptation.
In this adaptation mode, all of the proposed methods slightly improve the performances of the speaker-independant system compared with *SMLLR* and Structural EigenVoices technique whatsoever the available amount of adaptation data. The techniques *EV→SMLLR* and *SMLLR→EV* give similar results, as do the technique *SEV→SMLLR* compared to *SMLLR→SEV*. Thus it seems that the order of combination of *SMLLR* with one of the EigenVoices-based technique does not influence the quality of the generated adapted models.

	1	10	50	100	300	600
SMLLR	88.6	90.7	91.1	91.5	92.3	92.4
EV	**89.8**	90.0	90.1	90.0	90.1	90.1
SEV	**89.8**	90.7	91.3	91.1	91.1	91.4
SMLLR→EV	**89.8**	90.4	91.2	92.1	92.8	92.8
SMLLR→SEV	**89.8**	**91.0**	91.5	91.9	92.8	**92.9**
EV→SMLLR	**89.8**	90.6	**91.8**	**92.2**	92.6	92.8
SEV→SMLLR	**89.8**	90.7	91.7	**92.2**	**92.9**	**92.9**

Table 1. Comparison of the proposed approaches with *SMLLR*, Structural EigenVoices (SEV) and EigenVoices (EV) in supervised batch mode

The table 2 shows the results of the proposed approaches compared with *SMLLR*, EigenVoices technique and Structural EigenVoices technique for an unsupervised incremental adaptation.
The techniques where *SMLLR* adaptation is followed by an EigenVoices-based adaptation are significantly more powerful than techniques which do the opposite. We explain this behaviour by the

fact that EigenVoices can constrain too much the adapted models used later by *SMLLR*. As *SMLLR* has no effect with the two first utterances, the adapted models generated by EV or SEV may be located in a bad portion of the speaker-space. This could explain the poor results of the approaches *EV→SMLLR* and *SEV→SMLLR*.

	1	5	10	15	20	30
SMLLR	88.6	90.4	90.5	90.4	90.6	**91.0**
EV	**89.8**	90.0	90.0	90.0	90.0	90.1
SEV	**89.8**	90.5	**90.8**	**90.9**	**91.0**	90.9
SMLLR→EV	**89.8**	90.8	90.7	90.2	90.9	89.7
SMLLR→SEV	**89.8**	**90.9**	**90.8**	90.3	90.8	**91.0**
EV→SMLLR	89.1	89.8	90.0	90.4	90.6	**91.0**
SEV→SMLLR	88.9	89.4	89.9	90.1	90.0	90.8

Table 2. Comparison of the proposed approaches with *SMLLR*, Structural EigenVoices (SEV) and EigenVoices (EV) in unsupervised incremental mode

6. CONCLUSION

We have proposed in this paper a structural version of EigenVoices technique and four straightforward methods which combine *SMLLR* and EigenVoices-based techniques for speaker adaptation in both supervised batch mode and unsupervised incremental mode. It has been shown experimentaly that Structural EigenVoices can push bask the early saturation in performance encountered by the regular version of EigenVoices technique. Besides, for a supervised batch adaptation, the four proposed methods improve the performances of an ASRS over both *SMLLR* and EigenVoices-based techniques whatever the available amount of adaptation data. For a unsupervised incremental adaptation, *SMLLR→SEV* seemed to provide the best results compared to the other methods which were evaluated.

7. REFERENCES

[1] C.J. Leggetter and P.C. Woodland, "Maximum Likelihood Linear Regression for Speaker Adaptation of Continous Density Hidden Markov Models," *Comp. Speech and Language*, vol. 9, pp. 171–185, 1995.

[2] C.J. Leggetter and P.C. Woodland, "Flexible Speaker Adaptation using Maximum Likelihood Linear Regression," *Eurospeech'1995*, pp. 1155–1158, 1995.

[3] R. Kuhn, P. Nguyen, J.-C. Junqua, and al., "Eigenvoices for Speaker Adaptation," *ICSLP'1998*, 1998.

[4] R. Kuhn, P. Nguyen, J.-C. Junqua, R. Boman, N. Niedzielski, S. Fincke, K. Field, and M. Contolini, "Fast Speaker Adaptation using A Priori Knowledge," *ICASSP'1999*, pp. 1587–1590, 1999.

[5] R. Kuhn, J.-C. Junqua, P. Nguyen, and N. Niedzielski, "Rapid Speaker Adaptation in Eigenvoice Space," *IEEE Trans. Speech Audio Proc.*, vol. 8, no. 6, pp. 695–707, 2000.

[6] H. Botterweck, "Very Fast Adaptation for Large Vocabulary Continuous Speech Recognition using Eigenvoices," *ICSLP'2000*, pp. 354–357, 2000.

[7] K.-T. Chen, W.-W. Liau, H.-M. Wang, and L.-S. Lee, "Fast Speaker Adaptation using Eigenspace-based Maximum Likelihood Linear Regression," *ICSLP'2000*, pp. 742–745, 2000.

[8] N.J.-C. Wang, S. S.-M. Lee, F. Seide, and L.-H. Lee, "Rapid Speaker Adaptation using A Priori Knowledge by Eigenspace Analysis of MLLR Parameters," *ICASSP'2001*, 2001.

[9] P. Nguyen, "Fast Speaker Adaptation," Tech. Rep., Speech Technology Laboratory, 1998.

[10] D. Fohr, O. Mella, and C. Antoine, "The Automatic Speech Recognition Engine ESPERE : Experiments on Telephone Speech," *ICSLP'2000*, pp. 246–249, 2000.

[11] K. Shinoda and C.-H. Lee, "A Structural Bayes Approach to Speaker Adaptation," *IEEE Trans. Speech Audio Proc.*, vol. 9, no. 3, pp. 276–287, 2001.

[12] V.V. Digalakis, "Online Adaptation Hidden Markov Models using Incremental Estimation Algorithms," *IEEE Trans. Speech Audio Proc.*, vol. 7, no. 3, pp. 253–261, 1999.