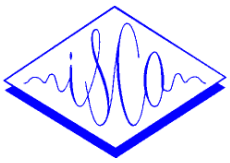

6th International Conference on Spoken Language Processing (ICSLP 2000)
Beijing, China
October 16-20, 2000

STRUCTURAL *MAXIMUM A POSTERIORI* LINEAR REGRESSION FOR UNSUPERVISED SPEAKER ADAPTATION

Tor André Myrvoll * *Olivier Siohan* *Chin-Hui Lee* *Wu Chou*

Multimedia Communications Research Lab
Bell Laboratories – Lucent Technologies, 600 Mountain Ave., Murray Hill, NJ 07974, USA
www.multimedia.bell-labs.com

ABSTRACT

In this paper we introduce an approach to transformation based model adaptation techniques. Previously published schemes like MLLR define a set of affine transformations to be applied on clusters of model parameters. Although it has been shown that this approach can yield good results when adaptation data is scarce, an inherent problem needs to be considered: the number of transformations used has a significant influence on the adaptation performance. Using too many transformations will result in poorly estimated transformation parameters, eventually leading to a model that overfits the adaptation data. On the other hand, when too few transformations are used, a restricted mapping is obtained, leading to a suboptimal adapted model. We address this problem by estimating the transform parameters in a *maximum a posteriori sense*, using a set of hierarchical priors arranged in a tree structure. We show that this approach yields a significant improvement compared to MLLR when doing unsupervised model adaptation on the WSJ spoke 3 test.

1. INTRODUCTION

It is well known that the performance of automatic speech recognition systems is sensitive to mismatches between training and testing conditions, typical mismatches being channel and speaker variations[1]. Several different approaches to this problem has been put forward, including preprocessing of the speech signal, adaptation of the model parameters and robust decision strategies. In this paper we will consider a model adaptation technique.

In general, given a hidden Markov model (HMM) specified by its parameters Λ and some relevant adaptation data $\mathbf{X}$, we obtain a new model given by the parameters $\tilde{\Lambda} = F(\Lambda, \mathbf{X})$, where $F(\cdot,\cdot)$ is some predefined mapping. For practical reasons it is commonplace to split the model parameters into M disjoint clusters, $\lambda_1, \ldots, \lambda_M$, and then associate a unique mapping $F_{\eta_m}(\cdot)$ to each cluster m. Here η_m can either be an index or some parameter set specifying the mapping. In the case of a parametric mapping $F_\eta(\cdot)$, the model adaptation problem now becomes to estimate the parameters η. This is often done in the *maximum likelihood* (ML) sense:

$$\hat{\eta}_{ML} = \underset{\eta}{\operatorname{argmax}}\, p(\mathbf{X}|\eta, \Lambda). \tag{1}$$

Another approach is to use the *maximum a posteriori* (MAP) formulation. Here our prior knowledge about the plausibility of the different values of η is expressed as a *prior distribution* $p(\eta|\phi)$, where ϕ is known as *hyperparameters*. The MAP estimation problem can now be formulated as:

$$\begin{aligned}\hat{\eta}_{MAP} &= \underset{\eta}{\operatorname{argmax}}\, p(\eta|\mathbf{X}, \Lambda) \\ &= \underset{\eta}{\operatorname{argmax}}\, p(\mathbf{X}|\eta, \Lambda) p(\eta|\phi).\end{aligned} \tag{2}$$

Maximum likelihood linear regression (MLLR) is one of the best know implementations of the ideas expressed above. Here the parametric mapping takes the form of an *affine transformation* of the mean vectors of the Gaussian mixtures in the HMM,

$$\begin{aligned}\hat{\mu} &= A\mu + b \\ &= W\tilde{\mu}.\end{aligned} \tag{3}$$

where $W = [A\ b]$ and $\tilde{\mu}$ is the augmented mean vector. In the MLLR formulation the parameters $\eta = W$ are estimated using the ML formulation in equation (1). By applying the transformations on all the mean vectors in their respective clusters all parameters will be updated, thus allowing us to update mixtures not even observed in the possibly sparse adaptation data.

A naive implementation of MLLR has serious shortcomings though. Using a set of predetermined clusters allows for the possibility that too little data is available for a proper estimate to be made. This problem can be alleviated by choosing clusters dynamically, e.g. by arranging the clusters in a hierarchical tree structure and choosing which cluster to use according to a threshold on the amount of data available.

Another improvement is to specify a prior distribution, $p(W)$, for each of the transformations. This leads to the *maximum a posteriori linear regression)* (MAPLR) formulation [2]. Provided that good priors are chosen, the estimation of the transformations should be more robust. In this paper we will extend the MAPLR framework to make use of hierarchical priors. This new development enables us to choose better prior distributions, yielding robust and well behaved transformations using even very small amounts of adaptation data.

In the next section we will first explain the concept of hierarchical priors and make some assumptions regarding their parametric form as well as introducing the necessary approximations. Next we will show how we can arrange the priors in a tree structure that corresponds to a hierarchical tree of HMM parameter clusters, and then use this correspondence to estimate one transformation for each cluster in top down manner. Finally we will present some experimental results using unsupervised speaker adaptation on the WSJ Spoke 3 task.

* This work was done while T. A. Myrvoll was on leave from the Department of Telecommunications, Norwegian University of Science and Technology, Norway.

2. STRUCTURAL MAXIMUM A POSTERIORI LINEAR REGRESSION

2.1. Hierarchical priors

The use of hierarchical priors for model adaptation was first seen in [3], where it was used to model an acoustic mismatch probability density function (PDF). We will use a similar approach here, extending the algorithm to the MAPLR framework. The basic assumption is that each prior has a hyperprior, each hyperprior a hyper-hyperprior, and so on. As in [3] all the priors in the hierarchy will have the same parametric form, and will be estimated in a recursive manner according to their relative position in a tree structure. Informally, what we want to do is the following:

1. Consider a subset λ_1 of the HMM parameters Λ[1]. Let W_1 be the transformation to be applied to the mean vectors in λ_1. Using a prior $p(W_1)$ and some data $\mathbf{X}$, we find the MAP estimate of W_1,
$$\begin{aligned}\hat{W}_1 &= \underset{W'}{\operatorname{argmax}} p(W'|\mathbf{X}, \lambda_1) \\ &= \underset{W'}{\operatorname{argmax}} p(\mathbf{X}|W', \lambda_1) p(W').\end{aligned} \tag{4}$$
2. Now consider a subset $\lambda_2 \subset \lambda_1 \subset \Lambda$. Using W_1's posterior distribution, $p(W_1|\mathbf{X}, \lambda_1)$, as our new prior distribution we can find a MAP estimate for the transformation W_2 to be applied to the mean vectors in λ_2.

Using the steps above in a recursive manner we obtain a set of transformations that is estimated in a MAP sense using a relevant prior distribution. The exact form of the prior and posterior distributions, as well as a way of constructing the set of parameter clusters will be discussed below.

As always when the Bayesian paradigm is involved, the choice of priors is a delicate one. In this work we assume that a transformation W has a normal distribution with mean M and covariance Φ. An immediate problem with this choice of prior is that the posterior distribution $p(W|X, \Lambda)$ will not be contained in a similar parametric family as $p(W)$, as is easily seen from the following expression:

$$p(W|\mathbf{X}, \Lambda) = \frac{\sum_{S \in \mathcal{S}} \sum_{L \in \mathcal{L}} p(\mathbf{X}, S, L|W, \Lambda) p(W)}{p(\mathbf{X})}. \tag{5}$$

Here $\mathcal{S}$ and $\mathcal{L}$ are the sets of feasible state and mixture sequences respectively. Using the exact form in (5) is clearly infeasible as the number of terms in the expression increases by an exponential rate from one level of priors to the next. Approximations will have to be made, and in this work we approximate $p(W|\mathbf{X}, \Lambda)$ by a normal distribution with a mean equal to the MAP estimate $\hat{W}$ and the same covariance as the prior $p(W)$.

Applying the two steps specified above using the normality assumption and the approximation of the posterior by a normal distribution, we obtain a sequence of prior distributions, $(p(W_i|W_{i-1}))_i$, where

$$p(W_i|W_{i-1}) \propto \mathcal{N}(W_i; W_{i-1}, \Phi). \tag{6}$$

In the next subsection we show how to arrange these priors into trees.

2.2. A tree structure of transforms

Allowing several priors to share the same hyperprior gives us the tree structure in Figure 1. We now want to combine the hierarchical tree structure of priors with a corresponding structure of the HMM parameters. As we will be adapting the Gaussian mean vectors in this work, we will only be concerned with clusters of Gaussians.

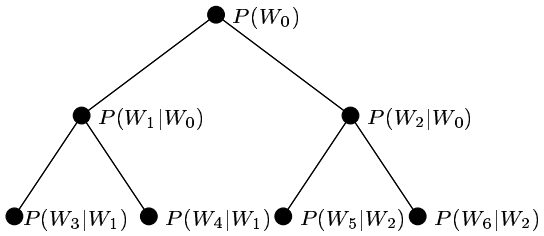

Figure 1: A hierarchical tree structure of priors.

A tree structure of parameter clusters is defined as follows. Let the top node contain all the relevant parameters Λ in our HMM. Now split this node into K separate child nodes, each containing a subset $\lambda_k \subset \Lambda$ so that $\lambda_k \cap \lambda_l = \emptyset,\ \forall\ k \neq l$ and $\bigcup_k \lambda_k = \Lambda$. This procedure can be repeated for each of the child nodes, eventually defining some tree $\mathcal{T}$. In practice this can be accomplished in one of several ways. On one hand, phonetic knowledge can be used to cluster states belonging to acoustically similar models. Another alternative is to define a measure of distance between two Gaussians and use this together with one of several well-known clustering algorithms. In this work we followed the approach in [4] and used a distortion measure based on the *Kullback-Leibler divergence*. Starting out with all the Gaussians in the HMM in the top node and then using our distortion measure and the K-means algorithm, we divide the node into K clusters. Doing this recursively to all K clusters gives us a tree of increasingly fine resolution.

We now assume that each node in the tree will yield a prior distribution that can be used by its child nodes. Starting with the top node we find the MAP estimate of a transformation matrix W_0 as described in [2]. This matrix is then propagated down the tree and used to define the prior for the next level of transformations. This process will terminate for one of two reasons: either a final node of the tree will be reached, or the amount of data available is considered insufficient to make a reliable estimate of the local transform W_k. Finally, when this process has terminated, each Gaussian in the HMM will be adapted using some local transform W. If a transform was successfully estimated in the tree node where the Gaussian resides, it will be used. If no transform could be estimated, the closest transform from some higher level in the tree will be applied. This is illustrated in Figure 2, where the circles filled with black illustrates successfully estimated transforms, and the white circles are nodes where too little data were available. In the next section we will show how to estimate the transform in a given node.

2.3. Estimating the transforms

The MAP estimate of the transform W_k in node k is not available as a closed form solution. However, the problem lends itself to the use of the *EM-algorithm* [5] which yields the following object function to be minimized:

$$\begin{aligned}Q(W_k, \hat{W}_k) &= \log p(\hat{W}_{k-1}) \\ &+ \sum_{S \in \mathcal{S}} \sum_{L \in \mathcal{L}} p(\mathbf{X}, S, L|\Lambda, W_k) \log p(\mathbf{X}, S, L|\Lambda, \hat{W}_k)\end{aligned} \tag{7}$$

[1] From here on we consider Λ to be the set of mean vectors from the Gaussian mixture distributions in the HMM only, as these are the only parameters that will be adapted in this work.

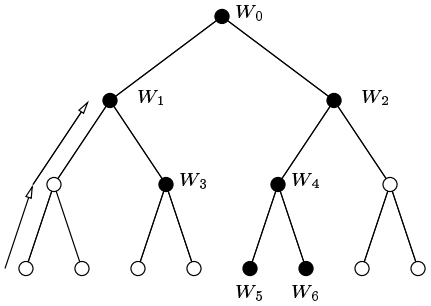

Figure 2: Adaptation using the closest transformation available.

As previously, $\mathcal{S}$ and $\mathcal{L}$ are the sets of feasible state and mixture sequences respectively. $p(\hat{W}_k|W_{k-1})$ is the prior propagated from the parent node.

Differentiating (7) with respect to every element in $\hat{W}_k$ and setting the result equal to zero gives us a set of linear equations that is equivalent to the MAPLR estimation problem presented in [2]. Using a full covariance matrix Φ for the prior results in a system of $p \times (p+1)$ equations in $p \times (p+1)$ unknowns, p being the mean vector dimension, which is clearly a considerable numerical problem. However, using a diagonal covariance matrix results in p sets of $p+1$ equations in $p+1$ unknowns, which is the same computational complexity as MLLR. For the derivation of the MAPLR equations, see [6].

A final note should be made on solving the linear equations. As we go down the tree the systems of equations to be solved becomes increasingly ill-conditioned, making standard approaches like LU-factorization behave poorly. Iterative methods based on the conjugate gradient algorithm seem to behave better, although care has to be taken with regard to the convergence[7].

3. EXPERIMENTS AND RESULTS

3.1. Database and baseline system

Experiments are performed on the 1993 Spoke3 test set of the WSJ task. The Spoke3 data consists of 10 non-native speakers of American English. Each speaker provided 40 utterances used for model adaptation and 40 utterances for testing. Model adaptation experiments have been carried out for each speaker using various amount of adaptation data, ranging from 1 to 40 utterances. In order to get statistically representative results, adaptation experiments are repeated for 3 different selections of the set of adaptation utterances (except for 40 utterances where a single set was used). For example, experiments involving 5 adaptation utterances have been done using 3 different sets of 5 adaptation utterances, randomly selected from the 40 utterances available in the adaptation data.

Triphone HMM models are built on the WSJ SI-84 training set using a decision-tree state tying algorithm [8]. A total of 3448 tied-states with an average of about 11 Gaussian mixture components per state is obtained. A 5K-word pronunciation lexicon was generated automatically using a general English text-to-speech system [9]. The language model used in the experiments is the standard trigram language model provided by NIST for the WSJ task.

A standard Mel frequency cepstral coefficient (MFCC) front-end is used to create a feature vector of 39 components, consisting of 12 MFCC component plus the log-energy term and their first and second derivatives. Cepstral mean normalization is applied on each sentence.

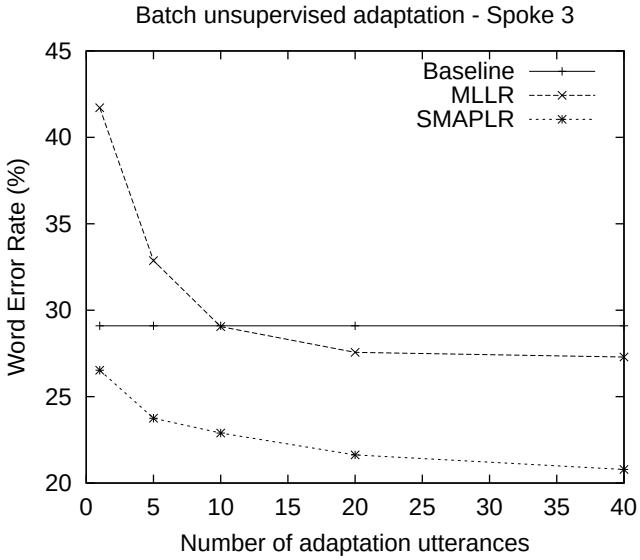

Figure 3: Word error rate (%) for batch unsupervised experiments for MLLR and SMAPLR for various amount of adaptation utterances.

The tree of Gaussian densities used for adaptation is designed so that the number of children in each node is at most 10, except for the last layer in the tree which contains 40 terminal (single Gaussian) nodes.

3.2. Experimental results

Two series of adaptation experiments have been performed. The first one is a series of *unsupervised batch* adaptation experiments: the acoustic models are adapted based on the adaptation utterances extracted from the adaptation data, and the adapted models are used to recognize the test data. The second series of experiments is based on *unsupervised online* adaptation, sometimes called auto-adaptation or self-adaptation: while recognizing the test data, the acoustic models are periodically updated using the previously recognized utterances to estimate the transformation parameters.

In the unsupervised batch adaptation experiments, for each speaker, the adaptation data is recognized using the 5K lexicon and the obtained hypothesized transcriptions are used to carry out the adaptation. We should point out that the lexicon used to derive the hypothetical transcription of the adaptation data is the 5K-closed lexicon used for testing. Because the vocabulary of the adaptation utterances is different from the vocabulary used in the test set, many adaptation utterances can therefore contain out of vocabulary words. This is an especially difficult adaptation scenario since the hypothesized transcription of the adaptation data is likely to contains errors. According to our experimental design, both MLLR and SMAPLR use the same number of transformation matrices located at the same nodes in the tree (the same sets of adaptation utterances are also used). Therefore, the only difference between the 2 adaptation techniques is in the estimation criterion, maximum likelihood for MLLR, maximum a posterior with hierarchical priors for SMAPLR. Figure 3 represents experimental results for MLLR and SMAPLR adaptation for various amount of adaptation data. The results are given in terms of av-

Adapt. method	WER (%)
Baseline	29.1
MLLR	21.6
SMAPLR	19.7

Table 1: Word error rate (%) for unsupervised online adaptation experiments for MLLR and SMAPLR.

erage word error rate over the 10 speakers. It clearly appears that SMAPLR outperforms MLLR for any amount of adaptation data, as it was also observed in our previous experiments on supervised adaptation [10]. We should point out that no attempt was made to optimize the performance of MLLR nor SMAPLR by doing a careful selection of the number of transformation matrices for each amount of adaptation data. The threshold used to control the number of transformations was selected once for all, and never modified. The experimental setting is similar to the one used for batch supervised experiments described in [10]. Obviously, with this setting, MLLR clearly overfits the adaptation data. Because of the prior information, the SMAPLR estimates avoid overfitting and a reasonable generalization is obtained. One can argue that by reducing drastically the number of transformation matrices in MLLR, one might avoid overfitting, as it was experimentally shown in [10] for supervised adaptation. This illustrates that SMAPLR is much less sensitive than MLLR to implementation details like the selection of the number of transformations. Moreover, we believe that when the number of transformation matrices is carefully optimized, the best SMAPLR setting can still outperform the best MLLR scenario, for any amount of adaptation data. This was illustrated in [11] where finely tuned MAPLR and MLLR were compared, showing a systematic advantage of the Bayesian estimation over the maximum likelihood estimation.

In our online unsupervised adaptation experiments, the models are adapted periodically, every 3 test utterances, starting from the initial speaker independent models. Rather than using a complex online approach based on a recursive Bayesian formulation as in [12], we choose to keep 2 sets of acoustic models in memory: the original speaker independent model and the adapted model. The adapted model is used for recognition and to generate the state segmentation used for adaptation. The original speaker independent model is used to carry out the estimation of the transformation matrices using the sufficient statistics that have been accumulated up to the current test utterance. In Table 1, the online unsupervised adaptation results are given for MLLR and SMAPLR adaptation. Again, SMAPLR show some improvement over MLLR. It might seem surprising that the word error rate is close to what was obtained with batch unsupervised adaptation with 40 adaptation experiments (one should expect something worse than the 40 utterance unsupervised adaptation). It is important to remember that in our experiments, the batch adaptation utterances contain many out of vocabulary words, while the online adaptation uses a closed lexicon, explaining the relatively bad performance of the batch unsupervised adaptation compared to the online adaptation.

4. CONCLUSIONS

We have presented an extension to the MAPLR approach to model adaptation. Hierarchical priors are arranged in a tree structure, estimated in a top-down manner and used to find robust estimates for the transformation matrices used in a linear regression step. Compared to the original maximum likelihood based approach (MLLR), we obtain robust estimates over a range of available adaptation data with a minimum of parameter tuning. Because of the hierarchical prior information, the SMAPLR algorithm is less sensitive than MLLR to the number of transformations, introduces a dependency in the estimation of transformation matrices between nodes having a common parent, and provides a convenient way to derive a prior distribution $p(W)$ in each node of the tree (which was lacking in our original MAPLR formulation). Our unsupervised adaptation experiments on the 1993 Spoke3 test set of the WSJ task show the superiority of the proposed approach over MLLR.

5. REFERENCES

[1] C.-H. Lee, "On stochastic feature and model compensation approaches to robust speech recognition," *Speech Communication*, vol. 25, pp. 29–47, 1998.

[2] O. Siohan, C. Chesta, and C.-H. Lee, "Hidden Markov model adaptation using maximum a posteriori linear regression," in *Workshop on Robust Methods for Speech Recognition in Adverse Conditions*, (Tampere, Finland), 1999.

[3] K. Shinoda and C.-H. Lee, "Structural MAP speaker adaptation using hierarchical priors," in *Proc. of IEEE Workshop on Speech Recognition and Understanding*, 1997.

[4] K. Shinoda and C.-H. Lee, "A structural Bayes approach to speaker adaptation," *IEEE Transactions on Speech and Audio Processing*, 2000. To appear.

[5] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of Royal Statistical Society Ser. B*, vol. 39, pp. 1–39, 1977.

[6] O. Siohan, C. Chesta, and C.-H. Lee, "Joint maximum *a Posteriori* adaptation of transformation and HMM parameters." Submitted to IEEE Trans. on Speech and Audio Processing.

[7] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical recipes in C. The Art of Scientific Computing*. Cambridge University Press, 1988.

[8] W. Reichl and W. Chou, "A decision tree state tying based on segmental clustering for acoustic modeling," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, (Seattle, Washington, USA), pp. 801–804, 1998.

[9] R. W. Sproat and J. P. Olive, "Text-to-speech synthesis," *AT&T Technical Journal*, vol. 74, pp. 35–44, 1995.

[10] O. Siohan, T.-A. Myrvoll, and C.-H. Lee, "Structural maximum *a Posteriori* linear regression for fast HMM adaptation," in *Workshop on Automatic Speech Recognition: Challenges for the new Millenium*, (Paris, France), ISCA ITRW ASR2000, Sept. 2000.

[11] C. Chesta, O. Siohan, and C.-H. Lee, "Maximum a posteriori linear regression for hidden Markov model adaptation," in *Proceedings of European Conference on Speech Communication and Technology*, vol. 1, (Budapest, Hungary), pp. 211–214, 1999.

[12] S. Wang and Y. Zhao, "On-line Bayesian speaker adaptation using tree-structured transformation and robust priors," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, (Istanbul, Turkey), June 2000.