Discriminative Cluster Adaptive Training

Kai Yu and Mark J. F. Gales, *Member, IEEE*

***Abstract*—Multiple-cluster schemes, such as cluster adaptive training (CAT) or eigenvoice systems, are a popular approach for rapid speaker and environment adaptation. Interpolation weights are used to transform a multiple-cluster, canonical, model to a standard hidden Markov model (HMM) set representative of an individual speaker or acoustic environment. Maximum likelihood training for CAT has previously been investigated. However, in state-of-the-art large vocabulary continuous speech recognition systems, discriminative training is commonly employed. This paper investigates applying discriminative training to multiple-cluster systems. In particular, minimum phone error (MPE) update formulae for CAT systems are derived. In order to use MPE in this case, modifications to the standard MPE smoothing function and the prior distribution associated with MPE training are required. A more complex adaptive training scheme combining both interpolation weights and linear transforms, a structured transform (ST), is also discussed within the MPE training framework. Discriminatively trained CAT and ST systems were evaluated on a state-of-the-art conversational telephone speech task. These multiple-cluster systems were found to outperform both standard and adaptively trained systems.**

***Index Terms*—Cluster adaptive training (CAT), discriminative training, eigenvoices, minimum phone error (MPE), multiple-cluster HMM.**

I. Introduction

ADAPTIVE training is a widely used technique to build speech recognition systems on nonhomogeneous data. This type of data occurs in many scenarios, for example, broadcast news and conversational telephone speech. In adaptive training, a set of transformations is used to represent the unwanted acoustic variabilities [1]. A *canonical* model is constructed which only represents the underlying speech variability. This canonical model is then adapted to a particular test speaker or environment. There are two sets of parameters to be estimated during training: the canonical model and the set of transformations. To allow the standard decoding framework to be used, the final adapted model is usually required to be a standard hidden Markov models (HMMs). However, the form of the canonical model can vary depending on the nature of the transformation. One commonly used set of transformations is based on linear transforms, such as maximum likelihood linear regression (MLLR) [2] and constrained MLLR (CMLLR) [3]. In both cases, the canonical model is a set of standard HMMs. An alternative approach is based on a multiple-cluster canonical model. Here, multiple sets of HMMs, one for each cluster, are used. The adapted model is generated by interpolating the cluster parameters to form a standard HMM. The transformation in these cases is a set of interpolation weights, a weight for each cluster. Though this approach was originally motivated for rapid speaker adaptation, it can be effectively extended for generic adaptation in large vocabulary speech recognition [4]. To simplify training, these interpolation weights are usually only applied to the cluster means to yield the final, adapted, mean vector for each component. Cluster adaptive training (CAT) [5] and eigenvoices [6] are two such approaches. More discussions about adaptive training in multiple-cluster systems can be found in [4]. These multiple-cluster schemes are the approach investigated in this paper, in particular CAT is considered.

Maximum likelihood (ML) estimation of CAT systems has previously been published [5]. However, in state-of-the-art speech recognition systems, discriminative training is commonly employed to obtain the best performance. Discriminative training criteria take into account competing, incorrect, hypothesis in training. Unlike ML training, they make no assumption of model correctness. Commonly used criteria are maximum mutual information (MMI) [7], minimum classification error (MCE) [8], and minimum phone error (MPE) [9]. Of these criteria, MPE is currently popular for state-of-the-art speech recognition as it has been found to yield good performance [10]. One problem with using these discriminative criteria is that parameter optimization becomes more complex. Standard auxiliary functions based on expectation maximization (EM) cannot be directly used. To overcome this problem, the extended Baum–Welch algorithm (EBW) [11], [22] has been proposed. An alternative approach, which yields similar update formulae and is highly flexible, uses a *weak-sense auxiliary function* [10]. This is the approach adopted in this paper to derive discriminative update formulae for multiple-cluster systems. Discriminative training has previously been studied within the linear transform-based adaptive training framework [12]–[14].

In discriminative adaptive training, both model and transform parameters should be updated using the discriminative criterion. However, for some tasks, this is not necessarily the most appropriate approach. For example, in unsupervised test-set adaptation, where the correct transcription is not known, using discriminative techniques to directly estimate the test-set transform is not possible. If, during adaptive training, the transforms have been estimated using discriminative techniques, there will be a mismatch between the training and test configurations. To allow a consistent approach for transform estimation, a simplified form of discriminative adaptive training is often used.

Manuscript received July 28, 2004; revised June 16, 2005. This work was supported by DARPA under Grant MDA972-02-1-0013. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Li Deng.

K. Yu is with the Engineering Department, Cambridge University, Cambridge CB2 1PZ, U.K.

M. J. F. Gales is with Emmanuel College, Cambridge **[AUTHOR: PLEASE PROVIDE POSTAL CODE]**, U.K.

Digital Object Identifier 10.1109/TSA.2005.858555

Rather than discriminatively estimating both the model parameters and the transformations during training, only the model parameters are discriminatively trained. The transformations are estimated using the ML criterion and fixed for all subsequent discriminative training iterations [14]. The ML criterion is then used to find the transformation for the test set adaptation. For details of the discriminative estimation of the CAT interpolation weights, see [15].

In this paper, discriminative training for multiple-cluster model is investigated. In particular, MPE training is applied to state-of-the-art CAT systems. In order to successfully apply MPE training to a large vocabulary system, it is necessary to specify appropriate smoothing functions and prior, I-smoothing, distributions [10]. For these multiple-cluster systems, the forms of these functions must be modified so that they may be applied to all canonical model parameters. In addition, a more complex multiple-cluster model based discriminative adaptive training technique is discussed, which uses a combination of interpolation weights, as in CAT, and CMLLR to represent complex nonspeech variabilities. This will be referred to as a structured transform (ST) [16].

This paper is organized as follows. Section II introduces ML estimation for CAT systems. Section III describes standard MPE training and Section IV extends this to multiple-cluster systems. Obtaining good prior distribution and smoothing constants is discussed in Section V. Discriminative adaptive training with STs is then described in Section VI. Section VII presents experiments on a state-of-the-art conversational telephone speech task.

II. Cluster Adaptive Training

CAT [17] is a multiple-cluster HMM training approach. The basic idea is to build a target-domain-specific mean vector for each component by using a weighted sum of multiple sets of mean vectors. A weight vector is computed for each distinct speaker[1] during training. The transform for test set adaptation is then a weight vector for each test speaker. There are two sets of model parameters that must be trained for the CAT system. The first are the canonical model parameters. Each component, m, of the canonical model consists of a prior, $c^{(m)}$, a covariance matrix (usually diagonal), $\mathbf{\Sigma}^{(m)}$, and a set of P means, one for each of the P clusters, normally arranged into a matrix $\mathbf{M}^{(m)}$

$$\mathbf{M}^{(m)} = \begin{bmatrix} \boldsymbol{\mu}_1^{(m)} & \dots & \boldsymbol{\mu}_P^{(m)} \end{bmatrix}.$$

The complete canonical model $\mathcal{M}$ consists of[2]

$$\mathcal{M} = \left\{ \{\mathbf{M}^{(1)}, \dots, \mathbf{M}^{(M)}\}, \{\mathbf{\Sigma}^{(1)}, \dots, \mathbf{\Sigma}^{(M)}\} \right\}$$

where M is the total number of components. The second set of parameters are the interpolation weights for each speaker, s, $\boldsymbol{\lambda}^{(s)}$

$$\boldsymbol{\lambda}^{(s)} = \begin{bmatrix} \lambda_1^{(s)} & \dots & \lambda_P^{(s)} \end{bmatrix}^T \tag{1}$$

where $\lambda_p^{(s)}$ is the interpolation weight for cluster p. In some systems a bias cluster is used where $\lambda_P^{(s)} = 1$ for all speakers. This bias cluster naturally occurs in an eigenvoice initialized system [5]. The adapted mean for a particular speaker s can then be written as

$$\boldsymbol{\mu}^{(sm)} = \mathbf{M}^{(m)} \boldsymbol{\lambda}^{(s)}. \tag{2}$$

ML CAT training has previously been published [5]. As in standard HMM training, an EM algorithm is employed. However, rather than simultaneously updating the transform parameters and the canonical model, the two updates are interleaved. The canonical model is estimated given a set of transforms, the transforms are then estimated for the new canonical model. At each iteration, the likelihood of the training data is guaranteed not to decrease. The ML update formulae for the canonical model are reproduced here as the discriminatively trained formulae are closely related to them. If diagonal covariance matrices are used,[3] then the ML-update formulae are given by

$$\mathbf{M}^{(m)T} = \mathbf{G}_{ml}^{(m)-1} \mathbf{K}_{ml}^{(m)} \tag{3}$$

$$\mathbf{\Sigma}^{(m)} = \frac{1}{\gamma_{ml}^{(m)}} \mathrm{diag}\left(\mathbf{L}_{ml}^{(m)} - \mathbf{M}^{(m)} \mathbf{K}_{ml}^{(m)} \right) \tag{4}$$

where the sufficient statistics are

$$\mathbf{G}_{ml}^{(m)} = \sum_{s,t} \gamma_{ml}^{(m)}(t) \lambda^{(s)} \lambda^{(s)T} \tag{5}$$

$$\mathbf{K}_{ml}^{(m)} = \sum_{s,t} \gamma_{ml}^{(m)}(t) \lambda^{(s)} \mathbf{o}(t)^T \tag{6}$$

$$\mathbf{L}_{ml}^{(m)} = \sum_{s,t} \gamma_{ml}^{(m)}(t) \mathbf{o}(t) \mathbf{o}(t)^T \tag{7}$$

$$\gamma_{ml}^{(m)} = \sum_{s,t} \gamma_{ml}^{(m)}(t) \tag{8}$$

and $\gamma_{ml}^{(m)}(t)$ is the posterior probability of being in component m at time t given the current canonical model parameters and transforms. For this work, the interpolation weights are only estimated using ML as in [5], so they are not discussed in detail.

When training a model using the EM algorithm, initialization is an important issue. For this work, the interpolation weights are initialized, rather than the model parameters. This allows the initialization of state-of-the-art speech recognition systems to be simply done [5], [15]. Once the interpolation weights are known, the component posterior $\gamma_{ml}^{(m)}(t)$ can be obtained from an appropriate large vocabulary standard system and (5)–(8) are used to update the canonical model. Two kinds of weight initialization are considered in this paper. The first is *cluster-based*, where each distinct speaker is assigned to a single cluster. This may either be done automatically, or using information from the training data such as gender or corpus. The second

[1] The term speaker in this paper could be replaced by acoustic environment or channel for example.

[2] In this paper, the Gaussian component priors and transition matrices are not included as their estimation, both ML-based and MPE-based, are similar to the standard estimation schemes. Refer to [18] for standard ML-based scheme and [10] for MPE-based scheme.

[3] This assumption is also applicable to discriminative CAT in later sections

form of initialization is based on *eigen-decomposition*. Here, simple models, such as monophones, are generated for each of the acoustic factors and an appropriate dimension eigenvoice system [6] is generated. The initial points for each speaker in this eigenspace is then used to initialize the weights. See [5] for more details.

Both CAT [17] and eigenvoices [19] are multiple cluster schemes. It is interesting to briefly contrast the two. Both systems use a set of distinct mean vectors. The "eigenvoices' correspond to the cluster mean matrices in CAT model. An eigenvoices system always employs an eigen-decomposition initialization approach based on PCA or LDA, but CAT systems may also employ prior knowledge for initialization. For some eigenvoices systems, the initialized basis eigenvoices are not further updated but directly used in adaptation and decoding [6], while CAT systems always update the multiple-cluster model [5]. If eigenvoices are updated using the maximum likelihood eigenspace (MLES) approach [20], it is equivalent to updating CAT cluster mean matrices and leaving covariance matrices unchanged [5]. Due to this close relationship, the discriminative training for CAT in this paper can also be used in eigenvoice systems.

III. Minimum Phone Error Training

The ML criterion aims to construct a model that maximizes the likelihood of the training data. However, in a classification task, the aim is to train a model that minimizes the error rate. Under various conditions, the ML criterion will also minimize the error rate [7]. However, these conditions are not satisfied for current speech recognition systems. This has led to an interest in training criteria that are more closely related to word error rate. These discriminative training criteria have become increasingly popular for speech recognition [7]–[9]. They take into account competing, incorrect hypothesis in training, rather than only the correct hypothesis. Thus, in addition to minimizing how "close" the model is to the correct hypothesis, it maximizes the "distance" from incorrect hypothesis. MPE has been found to yield good performance on state-of-the-art speech recognition tasks [9]. In this section, the MPE criterion is briefly described along with how it is applied to train standard HMMs.

The MPE criterion $\mathcal{R}(\mathcal{M})$ may be expressed as below [9], [10]

$$\mathcal{R}(\mathcal{M}) = \log\left(\frac{\sum\limits_{w} p(\mathbf{O}|\mathcal{M}_w)^{\kappa} P(w)\mathcal{A}(w)}{\sum\limits_{w} p(\mathbf{O}|\mathcal{M}_w)^{\kappa} P(w)}\right) + \log(p(\mathcal{M})) \tag{9}$$

where $\mathcal{A}(w)$ is a measure of the number of phones accurately transcribed, $\mathcal{M}_w$ is the composite model for word sequence w, κ is an acoustic deweighting factor commonly used in discriminative training, and $p(\mathcal{M})$ is the prior for the model parameters which is used to improve robustness of the estimates.

In MPE training, it is not possible to use EM to train the model parameters. This leads to the introduction of the EBW algorithm for MMI training [11], [22]. In this paper, a modified approach based on a *weak-sense* auxiliary function is used [10]. A weak-sense auxiliary function for $\mathcal{R}(\mathcal{M})$ around the *current model parameters* $\hat{\mathcal{M}}$ is a smoothing function $\mathcal{Q}^{mpe}(\mathcal{M};\hat{\mathcal{M}})$ such that

$$\left.\frac{\partial \mathcal{Q}^{mpe}(\mathcal{M};\hat{\mathcal{M}})}{\partial \mathcal{M}}\right|_{\mathcal{M}=\hat{\mathcal{M}}} = \left.\frac{\partial \mathcal{R}(\mathcal{M})}{\partial \mathcal{M}}\right|_{\mathcal{M}=\hat{\mathcal{M}}}. \tag{10}$$

Maximizing $\mathcal{Q}^{mpe}(\mathcal{M};\hat{\mathcal{M}})$ with respect to $\mathcal{M}$ does not guarantee to increase the objective function $\mathcal{R}(\mathcal{M})$. However, if $\mathcal{Q}^{mpe}(\mathcal{M};\hat{\mathcal{M}})$ reaches a local maximum at $\hat{\mathcal{M}}$, i.e., the gradient is 0 at that point, $\mathcal{R}(\mathcal{M})$ is guaranteed to also be at a local maximum. To increase the stability of the optimization, a smoothing function is required.

An appropriate weak-sense auxiliary function for standard HMMs is defined in [10]

$$\begin{aligned}\mathcal{Q}^{mpe}(\mathcal{M};\hat{\mathcal{M}}) = \mathcal{Q}^{n}(\mathcal{M};\hat{\mathcal{M}}) - \mathcal{Q}^{d}(\mathcal{M};\hat{\mathcal{M}})&\\ +\mathcal{F}(\mathcal{M};\hat{\mathcal{M}}) + \log(p(\mathcal{M}))&\end{aligned} \tag{11}$$

where the numerator and denominator auxiliary functions, $\mathcal{Q}^{n}(\mathcal{M};\hat{\mathcal{M}})$ and $\mathcal{Q}^{d}(\mathcal{M};\hat{\mathcal{M}})$, respectively, have the same form as the standard ML auxiliary function. The only difference between the numerator, denominator, and the standard auxiliary functions is that the "posterior probability" for Gaussian component m at time t, $\gamma_n^{(m)}(t)$ and $\gamma_d^{(m)}(t)$, are not calculated based on the correct transcription as the standard forward-backward algorithm does [18]. Instead, the forward-backward algorithm is first applied within each phone arc of the lattice.[4] It is then applied at word-level within the lattice to figure out the word arc posteriors. Phone accuracy is measured for each phone arc. The arcs with higher accuracy than the average are classified as numerator, and those with lower accuracy as denominator. $\gamma_n^{(m)}(t)$ is then calculated based on the numerator arcs by multiplying together the within arc component posterior occupancy, the word posterior occupancy and the phone-accuracy difference between the phone arc and the average accuracy. Similarly, $\gamma_d^{(m)}(t)$ is computed based on denominator arcs. Details can be found in [10].

The smoothing function $\mathcal{F}(\mathcal{M};\hat{\mathcal{M}})$ is required to ensure a convex weak-sense auxiliary function and consequently improve stability in optimization, this function must satisfy the following constraint

$$\left.\frac{\partial}{\partial \mathcal{M}} \mathcal{F}(\mathcal{M};\hat{\mathcal{M}})\right|_{\hat{\mathcal{M}}} = 0 \tag{12}$$

to ensure that the resulting auxiliary function is still a valid weak-sense auxiliary function for $\mathcal{R}(\mathcal{M})$. For the standard HMM, a modified Gaussian centred on the *current model parameters* is used [10].

To increase the robustness of the parameter estimates, a prior is used in the criterion (9) and, consequently, appears in the auxiliary function (11) [10]. The prior distribution over the model parameters may also be viewed as an additional smoothing function, referred to as "I-smoothing" [9]. Usually,

[4] In common with the majority of discriminative training schemes, lattices are used to represent possible denominator paths. These lattices are phone-marked before training. [9], [10]

a Normal–Wishart distribution is employed as in maximum *a posteriori* (MAP) training [21]. The prior parameters of $p(\mathcal{M})$ can be either based on ML estimates, which is the standard form [22], or based on MAP estimates, which is called MPE-MAP [23].

Differentiating the weak-sense auxiliary function (11) with respect to standard model parameters yields closed-form solution for standard MPE training [10]. The update formulae have the same form as the EBW algorithm [11], [22].

IV. MPE Training of Multiple-Cluster Model

The previous section has described the general form for the MPE criterion and the functions used to train standard HMMs. This section will discuss how MPE training can be applied to multiple-cluster models. The general form of the weak-sense auxiliary function in (11) can again be used. However, the form of the individual functions will change to reflect the multiple-cluster nature of the canonical model. For all estimates in this section, the transform parameters for each speaker $\boldsymbol{\lambda}^{(s)}$ are assumed to be known and fixed.

A. Numerator and Denominator Auxiliary Functions

The numerator and denominator functions for estimating the standard HMM parameters have the same form as the auxiliary function used for EM training. The same concept may be applied to multiple-cluster systems. The standard ML auxiliary function for CAT may be written as (ignoring terms independent of the model parameters)

$$\mathcal{Q}^{ml}(\mathcal{M};\hat{\mathcal{M}}) = -\frac{1}{2}\sum_{s,m,t}\gamma_{ml}^{(m)}(t)\Big\{\log|\mathbf{\Sigma}^{(m)}| + \Big(\mathbf{o}(t) - \mathbf{M}^{(m)}\boldsymbol{\lambda}^{(s)}\Big)^T\mathbf{\Sigma}^{(m)-1}\Big(\mathbf{o}(t) - \mathbf{M}^{(m)}\boldsymbol{\lambda}^{(s)}\Big)\Big\}. \tag{13}$$

It is possible to re-express this auxiliary function in terms of a set of ML sufficient statistics, $\mathbf{\Theta}_{ml}$, and a function over those statistics, $\mathcal{G}(\mathbf{\Theta}_{ml})$ such that

$$\mathcal{Q}^{ml}(\mathcal{M};\hat{\mathcal{M}}) = \mathcal{G}(\mathbf{\Theta}_{ml}) \tag{14}$$

where

$$\begin{aligned}\mathcal{G}(\mathbf{\Theta}_{ml}) = -\frac{1}{2}\sum_m\Big\{&\gamma_{ml}^{(m)}\log|\mathbf{\Sigma}^{(m)}| + \mathrm{tr}\Big(\mathbf{L}_{ml}^{(m)}\mathbf{\Sigma}^{(m)-1}\Big)\\ &- 2\mathrm{tr}\Big(\mathbf{K}_{ml}^{(m)}\mathbf{\Sigma}^{(m)-1}\mathbf{M}^{(m)}\Big)\\ &+ \mathrm{tr}\Big(\mathbf{G}_{ml}^{(m)}\mathbf{M}^{(m)T}\mathbf{\Sigma}^{(m)-1}\mathbf{M}^{(m)}\Big)\Big\}\end{aligned} \tag{15}$$

and

$$\mathbf{\Theta}_{ml} = \Big\{\gamma_{ml}^{(m)}, \mathbf{G}_{ml}^{(m)}, \mathbf{K}_{ml}^{(m)}, \mathbf{L}_{ml}^{(m)}\Big\}$$

as defined in (5)–(8). To obtain the numerator and denominator auxiliary functions, it is sufficient to replace the ML posterior, $\gamma_{ml}^{(m)}(t)$, in (5)–(8) with the appropriate numerator and denominator "posteriors" $\gamma_n^{(m)}(t)$ and $\gamma_d^{(m)}(t)$, respectively. This yields numerator and denominator statistics $\mathbf{\Theta}_n$ and $\mathbf{\Theta}_d$. The numerator auxiliary function can then be written in the general form as (15)

$$\mathcal{Q}^n(\mathcal{M};\hat{\mathcal{M}}) = \mathcal{G}(\mathbf{\Theta}_n) \tag{16}$$

and similarly for the denominator auxiliary function.

B. Smoothing Function

As discussed in Section III, the smoothing function $\mathcal{F}(\mathcal{M};\hat{\mathcal{M}})$ must yield the current current model $\hat{\mathcal{M}}$ parameters as a maximum to satisfy (12). Since the current parameters are dependent on the interpolation weights, one approach is to define a per speaker smoothing function, that satisfies (12) for each speaker. One suitable smoothing function is given by

$$\begin{aligned}\mathcal{F}(\mathcal{M};\hat{\mathcal{M}}) = -\sum_{m,s}\frac{D_m\nu_m^{(s)}}{2}&\\ \times\Big\{\log|\mathbf{\Sigma}^{(m)}| + \mathrm{tr}\Big(\hat{\mathbf{\Sigma}}^{(m)}\mathbf{\Sigma}^{(m)-1}\Big)&\\ + \Big(\mathbf{M}^{(m)}\boldsymbol{\lambda}^{(s)} - \hat{\boldsymbol{\mu}}^{(sm)}\Big)^T&\\ \times\mathbf{\Sigma}^{(m)-1}\Big(\mathbf{M}^{(m)}\boldsymbol{\lambda}^{(s)} - \hat{\boldsymbol{\mu}}^{(sm)}\Big)\Big\}&\end{aligned} \tag{17}$$

where $\hat{\boldsymbol{\mu}}^{(sm)} = \hat{\mathbf{M}}^{(m)}\boldsymbol{\lambda}^{(s)}$ and $\hat{\mathbf{\Sigma}}^{(m)}$ are the *current model parameters*. The constant D_m is a positive smoothing constant for component m to control the impact of smoothing function and make the optimization stable. This smoothing function is shown to be a valid smoothing function for all values of $\nu_m^{(s)}$ [15]. However, rather than using a constant value for all speakers, it is more appropriate to use this value to reflect the proportions of data for the particular component of a speaker.[5] In this work, it is set as

$$\nu_m^{(s)} = \frac{\sum_t\gamma_n^{(m)}(t)}{\sum_{s,t}\gamma_n^{(m)}(t)} \tag{18}$$

where the summation in the numerator only involves data associated with a particular speaker s.

This definition of the smoothing function is close to the standard auxiliary function. By doing a little algebra, it can be expressed in the same general form as (15), i.e.,

$$\mathcal{F}(\mathcal{M};\hat{\mathcal{M}}) = \mathcal{G}(\mathbf{\Theta}_s) \tag{19}$$

where

$$\mathbf{\Theta}_s = \Big\{D_m, D_m\mathbf{G}_s^{(m)}, D_m\mathbf{K}_s^{(m)}, D_m\mathbf{L}_s^{(m)}\Big\}$$

and

$$\mathbf{G}_s^{(m)} = \sum_s\nu_m^{(s)}\boldsymbol{\lambda}^{(s)}\boldsymbol{\lambda}^{(s)T} \tag{20}$$

[5] When using a constant value for $\nu_m^{(s)}$, the WER of MPE-CAT was about 0.1% worse compared to using (18) (with dynamic multiple-cluster ML prior and the same configurations as the 16 component development systems in Section VII)

$$\mathbf{K}_s^{(m)} = \mathbf{G}_s^{(m)} \hat{\mathbf{M}}^{(m)T} \tag{21}$$

$$\mathbf{L}_s^{(m)} = \hat{\boldsymbol{\Sigma}}^{(m)} + \hat{\mathbf{M}}^{(m)} \mathbf{G}_s^{(m)} \hat{\mathbf{M}}^{(m)T}. \tag{22}$$

For multiple-cluster MPE training, the selection of D_m is different from the single-cluster MPE training. This will be discussed in detail in Section V-B.

C. I-Smoothing Prior Distribution

The I-smoothing distribution used for a multiple-cluster HMM is a speaker-level Normal–Wishart distribution. Using the appropriate Normal–Wishart parameters, the general form of the log prior for multiple-cluster model parameters may be written as

$$\begin{aligned}\log p(\mathcal{M}) = &- \frac{\tau^I}{2} \sum_{s,m} \tilde{\nu}_m^{(s)} \\ &\times \left\{ \log |\boldsymbol{\Sigma}^{(m)}| + \mathrm{tr}\left(\tilde{\boldsymbol{\Sigma}}^{(m)} \boldsymbol{\Sigma}^{(m)-1} \right) \right. \\ &+ \left(\mathbf{M}^{(m)} \lambda^{(s)} - \tilde{\boldsymbol{\mu}}^{(sm)} \right)^T \\ &\left. \times \boldsymbol{\Sigma}^{(m)-1} \left(\mathbf{M}^{(m)} \boldsymbol{\lambda}^{(s)} - \tilde{\boldsymbol{\mu}}^{(sm)} \right) \right\}\end{aligned} \tag{23}$$

where τ^I is the parameter of the Normal–Wishart distribution which controls the impact of the prior. $\tilde{\boldsymbol{\mu}}^{(sm)}$ and $\tilde{\boldsymbol{\Sigma}}^{(m)}$ are the prior parameters for the mean and covariance matrix of the mth component for speaker s. $\tilde{\nu}_m^{(s)}$ is a slightly modified version of (18). Since in standard MPE training, ML estimates are often used as the priors, the ML posterior occupancy $\gamma_{ml}^{(m)}(t)$ are, therefore, used here to define $\tilde{\nu}_m^{(s)}$ instead of the numerator occupancy $\gamma_n^{(m)}(t)$.

This form of I-smoothing prior is similar to the smoothing function (17). The log prior may be expressed as

$$\log(p(\mathcal{M})) = \mathcal{G}(\boldsymbol{\Theta}_p) \tag{24}$$

where $\mathcal{G}()$ is the general form defined in (15)

$$\boldsymbol{\Theta}_p = \left\{ \tau^I, \tau^I \mathbf{G}_p^{(m)}, \tau^I \mathbf{K}_p^{(m)}, \tau^I \mathbf{L}_p^{(m)} \right\}$$

and

$$\mathbf{G}_p^{(m)} = \sum_s \tilde{\nu}_m^{(s)} \boldsymbol{\lambda}^{(s)} \boldsymbol{\lambda}^{(s)T} \tag{25}$$

$$\mathbf{K}_p^{(m)} = \sum_s \tilde{\nu}_m^{(s)} \boldsymbol{\lambda}^{(s)} \tilde{\boldsymbol{\mu}}^{(sm)T} \tag{26}$$

$$\mathbf{L}_p^{(m)} = \tilde{\boldsymbol{\Sigma}}^{(m)} + \left(\sum_s \tilde{\nu}_m^{(s)} \tilde{\boldsymbol{\mu}}^{(sm)} \tilde{\boldsymbol{\mu}}^{(sm)T} \right). \tag{27}$$

The global value τ^I shows the impact of the prior, which is experimentally determined as in standard MPE training [9]. In the experiments for this paper, the performance was found to be insensitive to the precise value of τ^I used (within a reasonable range). All experiments were using $\tau^I = 50$, the same value used for standard MPE training in [10].

Though the log prior (23) and the smoothing term (17) have similar forms, they are considered separately as they have different functions. The smoothing function $\mathcal{F}(\mathcal{M}; \hat{\mathcal{M}})$ is used to stabilize the optimization and control the update rate of MPE training. It is based on the *current model parameters* resulting in a similar form to the EBW re-estimation formulae [11], [22]. On the other hand, the prior distribution $\log(p(\mathcal{M}))$ is used to avoid over-training in a similar way to MAP training [21]. It ensures that for parameters with little data, the estimates are robust; hence, they are likely to have good generalization on unseen data. A range of possible priors, naturally not the current model parameters, can be used. Prior options for multiple-cluster MPE training will be discussed in Section V-A. Though, in theory, $\log(p(\mathcal{M}))$ with a large τ^I can also lead to stable optimization without the smoothing term $\mathcal{F}(\mathcal{M}; \hat{\mathcal{M}})$, it may lead to very slow update and the updated parameters may be dominated by the prior. Hence, in practice, the two smoothing functions (17) and (23) are both needed to achieve efficient, stable and robust MPE updates.

D. Update Formulae

All the elements of the MPE weak sense auxiliary function have been expressed in terms of a single function over a set of sufficient statistics

$$\mathcal{Q}^{mpe}(\mathcal{M}; \hat{\mathcal{M}}) = \mathcal{G}(\boldsymbol{\Theta}_n) - \mathcal{G}(\boldsymbol{\Theta}_d) + \mathcal{G}(\boldsymbol{\Theta}_s) + \mathcal{G}(\boldsymbol{\Theta}_p). \tag{28}$$

Differentiating this expression and equating to zero yields the following update formulae:

$$\mathbf{M}^{(m)T} = \mathbf{G}_{mpe}^{(m)-1} \mathbf{K}_{mpe}^{(m)} \tag{29}$$

$$\boldsymbol{\Sigma}^{(m)} = \frac{1}{\gamma_{mpe}^{(m)}} \mathrm{diag}\left(\mathbf{L}_{mpe}^{(m)} - \mathbf{M}^{(m)} \mathbf{K}_{mpe}^{(m)} \right) \tag{30}$$

where

$$\gamma_{mpe}^{(m)} = \gamma_n^{(m)} - \gamma_d^{(m)} + D_m + \tau^I \tag{31}$$

$$\mathbf{G}_{mpe}^{(m)} = \mathbf{G}_n^{(m)} - \mathbf{G}_d^{(m)} + D_m \mathbf{G}_s^{(m)} + \tau^I \mathbf{G}_p^{(m)} \tag{32}$$

$$\mathbf{K}_{mpe}^{(m)} = \mathbf{K}_n^{(m)} - \mathbf{K}_d^{(m)} + D_m \mathbf{K}_s^{(m)} + \tau^I \mathbf{K}_p^{(m)} \tag{33}$$

$$\mathbf{L}_{mpe}^{(m)} = \mathbf{L}_n^{(m)} - \mathbf{L}_d^{(m)} + D_m \mathbf{L}_s^{(m)} + \tau^I \mathbf{L}_p^{(m)}. \tag{34}$$

These can be combined to form the set of sufficient statistics

$$\boldsymbol{\Theta}_{mpe} = \left\{ \gamma_{mpe}^{(m)}, \mathbf{G}_{mpe}^{(m)}, \mathbf{K}_{mpe}^{(m)}, \mathbf{L}_{mpe}^{(m)} \right\}.$$

V. Prior Distribution and Smoothing Constant

The update formulae in the previous section have assumed that the form of the prior distribution and the smoothing constant for each component D_m are known. This section discusses the various forms of prior that may be used and the estimation of D_m for multiple-cluster systems.

A. Choice of Prior

One key issue in obtaining a "good" I-smoothing distribution is to choose an appropriate form for the prior distribution and associated parameters $\tilde{\boldsymbol{\mu}}^{(sm)}$ and $\tilde{\boldsymbol{\Sigma}}^{(m)}$. It is possible to use the same form of prior as the standard MPE training. In this *single-cluster* case, the prior for all speakers will be the same, normally specified as the ML-estimate [10]

$$\tilde{\boldsymbol{\mu}}^{(sm)} = \tilde{\boldsymbol{\mu}}^{(m)} = \frac{1}{\gamma_{ml}^{(m)}}\left(\sum_{t}\gamma_{ml}^{(m)}(t)\mathbf{o}(t)\right). \qquad (35)$$

In terms of the description of priors in this paper this is a single-cluster dynamic ML prior. However, more general forms of prior are possible for multiple-cluster systems. The prior does not have to be *dynamic*, namely the statistics accumulated on-the-fly. Parameters of an existing HMM model, for example a standard ML-SI model, can be used as the prior as well. This prior is fixed from iteration to iteration, called a *static* prior. Besides ML-SI model, a standard MPE-SI model can also be a valid prior, called a single-cluster static MPE prior. Similarly, accumulated MPE statistics for a standard HMM is a single-cluster dynamic MPE prior [15].

An interesting alternative is to use a *multiple-cluster prior*. Here, the speaker-specific mean prior $\tilde{\boldsymbol{\mu}}^{(sm)}$ in (23) is obtained by interpolating over a set of *multiple-cluster* priors for each speaker

$$\tilde{\boldsymbol{\mu}}^{(sm)} = \tilde{\mathbf{M}}^{(m)}\boldsymbol{\lambda}^{(s)} \qquad (36)$$

where $\tilde{\mathbf{M}}^{(m)}$ is the *multiple-cluster* prior mean matrix, $\boldsymbol{\lambda}^{(s)}$ is the interpolation weights for speaker s. In this approach, an existing CAT model may be used as a static prior. Alternatively, ML statistics for $\tilde{\mathbf{M}}^{(m)}$ and $\tilde{\boldsymbol{\Sigma}}^{(m)}$ can also be employed as dynamic priors. For example, substituting ML estimates for $\tilde{\mathbf{M}}^{(m)}$ and $\tilde{\boldsymbol{\Sigma}}^{(m)}$, the sufficient statistics (25)–(27) become

$$\boldsymbol{\Theta}_p = \left\{\tau^I, \frac{\tau^I}{\gamma_{ml}^{(m)}}\mathbf{G}_{ml}^{(m)}, \frac{\tau^I}{\gamma_{ml}^{(m)}}\mathbf{K}_{ml}^{(m)}, \frac{\tau^I}{\gamma_{ml}^{(m)}}\mathbf{L}_{ml}^{(m)}\right\}.$$

Comparing the above statistics with the ML statistics (5)–(7), they have been normalized by $\gamma_{ml}^{(m)}$ to yield "unit" counts, as the "occupancy" of I-smoothing part is represented by τ^I. This is a natural multiple-cluster extension from the standard single-cluster MPE training.

Examining the MPE sufficient statistics (31)–(34), when $\tau^I \rightarrow \infty$, sufficient statistics of I-smoothing distribution $\boldsymbol{\Theta}_p$ will dominate $\boldsymbol{\Theta}_{mpe}$. In this case, if a multiple-cluster prior is used, the MPE-CAT estimates will degrade to the multiple-cluster prior. In particular, if a multiple-cluster ML dynamic prior is used, the MPE estimates will degrade to the ML-CAT estimate (3) and (4).

B. Selection of the Smoothing Constant

The constant D_m is a critical value in MPE training to make the weak-sense auxiliary function convex and give rapid and stable update. Large value of D_m will guarantee that the MPE training does not go too aggressively to get stable update, but result in slow update. Small value may give fast update but affect the convexity of the weak-sense auxiliary function. There is no ideal approach for obtaining D_m satisfying both purposes. In common with the EBW updates, the value of D_m for weak-sense auxiliary function is set using empirically derived heuristics. As suggested in [10], in this work, D_m is determined by

$$D_m = \max\left(2\tilde{D}_m, E\gamma_d^{(m)}\right) \qquad (37)$$

where $\tilde{D}_m$ is the smallest value required to ensure the updated covariance matrix (30) is positive-definite, E is a user-specified constant, and $\gamma_d^{(m)}$ is the total denominator posterior occupancy for component m. To find an appropriate $\tilde{D}_m$, the equation $\boldsymbol{\Sigma}^{(m)} = \mathbf{0}$ must be solved from (30). For a single-cluster system, the equation for each dimension is a quadratic equation and can be easily solved [10]. However, for a multiple-cluster model, it can be shown that the order of the polynomial equation is $P+1$, where P is the number of clusters [15]. For some special cases, such as $P=2$ or $P=3$, i.e., cubic or quartic polynomial equation, closed-form solutions exist, and the largest real root can be found directly. For larger number of clusters, there are no simple closed-form solutions. Numerical approaches may be used to find the largest real root. Alternatively, as a rough approximation, $E\gamma_d^{(m)}$ may be used directly as D_m together with an appropriate variance floors to ensure the updated covariance matrices are positive definite.[6]

VI. Structured Transforms

Adaptive training normally uses a single transform to represent all nonspeech variabilities. However, for found data, there may be multiple acoustic factors affecting the speech signal. This motivates the use of multiple forms of transformations, denoted here as STs, to represent complex nonspeech variabilities in an adaptive training framework [16], [24]. In this paper, one particular form of ST is investigated which is appropriate for state-of-the-art speech recognition systems. Here, CAT interpolation weights [5] and CMLLR transforms [3] are combined together to form the transformation used for both adaptive training and test-set adaptation.

In the ST form, the canonical model is a multiple-cluster model. However, in addition to CAT, a speaker specific transformation of the feature space is also applied. Then, the models are trained in a transformed features space, where

$$\mathbf{o}^{(s)}(t) = \mathbf{A}^{(s)}\mathbf{o}(t) + \mathbf{b}^{(s)}(38)$$

$\mathbf{A}^{(s)}$ and $\mathbf{b}^{(s)}$ are the CMLLR transform for speaker s and the transformed mean is given by (2). Estimation of the multiple-cluster canonical model is a simple extension to the CAT approach [16]. The only modification to both ML and MPE training is that the model is estimated in the transformed feature-space, where the transformed feature vectors $\mathbf{o}^{(s)}(t)$ are used instead of standard feature vectors $\mathbf{o}(t)$.

The ML transformation estimation is implemented in a simple iterative process, where given the interpolation weights, the adapted mean, $\boldsymbol{\mu}^{(sm)}$ is used instead of $\boldsymbol{\mu}^{(m)}$ to estimate the CMLLR transform as described in [3]. Then, the interpolation weights, $\boldsymbol{\lambda}^{(s)}$ are estimated using the transformed features

[6] This approximate form has been shown in experiments to give only marginal difference from the exact form. In this work, the exact form is used for two-cluster systems and approximate selection for systems with more clusters.

$\mathbf{o}^{(s)}(t)$. Since a simplified MPE training scheme is employed here, discriminative re-estimation of transformation parameters are not considered.

VII. Experiments

A. Systems Description

Discriminative CAT was evaluated on a state-of-the-art large vocabulary speech recognition system, conversational telephone speech task. The training dataset consists of three corpora recorded with slight different acoustic conditions and collection framework. They are LDC Call-home English, Switchboard, Switchboard-Cellular, consisting of 5446 speakers (2747 female, 2699 male), about 295 h of data. The performance was evaluated on two held out test sets. A smaller development test set of half of the `dev01` test data, consisting of 59 speakers (30 female, 29 male), about 3 h, called the `dev01sub` test data. The second, larger, evaluation test dataset is the `eval03` dataset consisting of 144 speakers (77 female, 67 male), about 6 h. All systems used a 12-dimensional PLP front-end with log energy and its first, second and third derivatives with Cepstral mean and variance normalization. An HLDA transform was applied to reduce the feature dimension to 39. VTLN was also used. The use of simple adaptation schemes, mean, and variance normalization and VTLN, decreased the possible gains that could be obtained using adaptive training, but gave a more realistic baseline. A tri-gram language model was used in decoding.

Two kinds of systems were built. Both systems were built using the same state-clustered decision trees with 6189 distinct states. 16 components-per-state systems with four MPE training iterations were used for rapid development. 28 component-per-state systems with eight MPE training iterations were built for generating results of state-of-the-art systems. Unless otherwise stated, the standard form of I-smoothing using a dynamic single-cluster ML prior was used for all MPE training.

Gender-independent (GI) and gender-dependent (GD) MPE systems were built using standard MPE training technique. Due to insufficient training data, GD MPE system often gave poor performance [23]. To obtain good GD performance, a more complex I-smoothing taking into account the static prior information was used [23]. This system is referred to as GD MPE-MAP. It yielded the best possible performance of the state-of-the-art GD MPE systems. In this paper, gender labels were assumed to be known for both standard MPE GD and GD MPE-MAP systems in decoding.

Several adaptive MPE systems were constructed using the simplified MPE adaptive training. They include MPE-SAT, MPE-CAT and MPE-ST which employed CAT weights and CMLLR transforms as the STs. The MPE-SAT system employed standard MPE training technique and used CMLLR as the transformation [3]. During adaptive training and test adaptation, global interpolation weights were estimated for CAT and separate speech and silence transforms were used for CMLLR. To fully compare the adaptation performance, CMLLR adaptation was also applied on top of MPE baseline systems and MPE-CAT systems. For all systems, the test set supervision was generated using the MPE-SI models.

In the following experiments, pairwise significance test was done using NIST provided software `sctk-1.2`, where significance tests were using a standard approach [25].

B. Development Results

This section describes the initial development using the 16 Gaussian component per state system with a reduced number of MPE iterations.

Table I shows the baseline performance for ML and MPE training of the 16 component GI system. As expected the performance gain from MPE training is large, over 3% absolute for both tasks. Results for a two-cluster gender-initialized ML-CAT and GD systems are also given in Table I. The ML-CAT system was significantly better, using the pairwise test, than the ML-GI baseline for both test sets. It was slightly, not significantly, better than the ML-GD system in `dev01sub`, while significantly better in `eval03`.

I-smoothing is essential for obtaining good test set performance using MPE training [10]. The standard form of I-smoothing for training HMMs is to use a dynamic, single-cluster ML prior. As previously mentioned, the selection of the prior parameters is of additional interest for MPE-CAT systems as the number of model parameters to be estimated is greater than that of the equivalent standard HMM system. Thus, the form of prior used will have a greater influence than for the standard HMM system. A range of priors may be used, as described in Section IV-C. A single-cluster static MPE prior may be obtained from the standard MPE-GI model. Alternatively a dynamic MPE prior can be obtained from the single-cluster MPE statistics generated during training. Finally, a multiple-cluster dynamic ML prior can be obtained from the multiple cluster ML statistics during training. These three forms of prior, along with using a standard single-cluster ML prior were investigated.

Table II shows the development system performance of MPE-CAT using different I-smoothing prior distributions. All systems were initialized using the gender information. The form of the prior dramatically affects the error rate. For example, the performance on `dev01sub` varied from 29.3% upto 29.7%. It should be noted that all these values are better than the ML-CAT number of 32.6% and the MPE GI performance of 30.4% shown in Table I. The worst performance was obtained with standard dynamic ML prior. The best performance was obtained using either the static or dynamic single component MPE priors. The performance of the MPE-CAT system using either of the MPE-priors shown in Table II is significantly better, using the pairwise significance test, than both the ML-CAT system and the MPE-GI system. they were also significantly better than the complex GD MPE-MAP system.

There was little difference in performance between the dynamic and static MPE single cluster prior systems. However, since a dynamic MPE prior requires additional accumulates [15] (unless a bias cluster is used), all the following experiments of CAT and ST used the single-cluster static MPE prior.

A second interesting aspect for CAT systems is the number of clusters and how they are initialized. Again, the 16 component development system was used. Three forms of cluster initialization were investigated. The first two were cluster-based

TABLE I
16-COMPONENT ML AND MPE GI PERFORMANCE AND ML BASELINE PERFORMANCE FOR GD AND TWO-LUSTER CAT SYSTEMS

System	dev01sub		eval03	
	ML	MPE	ML	MPE
GI	33.4	30.4	32.6	29.2
GD	32.7	30.3	32.2	29.3
GD (MPE-MAP)	—	29.7	—	29.0
CAT	32.6	—	31.9	—

TABLE II
16-COMPONENT TWO-CLUSTER MPE-CAT SYSTEMS WITH DIFFERENT FORMS OF I-SMOOTHING PRIOR DISTRIBUTION

Baseline/Prior			Test sets	
Form	Type	Criterion	dev01sub	eval03
ML-CAT			32.6	31.9
MPE-GI			30.4	29.2
MPE-GD-MAP			29.7	29.0
multiple	dynamic	ML	29.7	28.9
single	dynamic	ML	29.7	29.1
		MPE	29.3	28.4
	static	MPE	29.3	28.5

TABLE III
16-COMPONENT SYSTEM WITH DIFFERENT INITIALISATION AND NUMBER OF CLUSTERS

Initialisation	Bias	#Clst	dev01sub		eval03	
			ML	MPE	ML	MPE
gender	no	2	32.6	29.3	31.9	28.5
corpus		3	32.3	29.2	31.7	28.3
eigen		3	32.3	29.0	31.5	28.2
eigen	yes	2	32.8	29.3	32.0	28.5
		3	32.3	29.0	31.6	28.3
		4	32.3	29.0	31.5	28.3

schemes where, the interpolation weights were initialized using either gender information, or the corpus information. The third form of initialization used an eigen-decomposition, as described in Section II. For the eigen-decomposition initialized systems, the bias cluster interpolation weight was either constrained to stay at one, called a bias cluster system or allowed to vary after initialization, called a no bias cluster system.

Table III shows the performance with different numbers of clusters and initialization. Initially examining the form of the initialization with no bias, the use of a three-cluster eigen-decomposition system was significantly better, using the pairwise test, than the two-cluster gender initialized for both ML and MPE training, though there was no significant difference between the eigen-initialized system and the corpus initialized system. For the eigen-initialized scheme, various systems using a bias were also constructed. For these systems, the use of three or four clusters was better than the two-cluster system. However, there was no significant difference in performance between any of the three or four cluster systems.

It is interesting to contrast the forms of system used here, where there are relatively few clusters, with the large number of clusters used in many eigenvoice systems. Many eigenvoice systems [6] use large numbers of clusters, but with relatively simple acoustic models, for example, single Gaussian component per-state models. These simple models are not appropriate for LVCSR. More complex systems have been built using ML eigenspaces [26]. However, on the same task, and starting from a better baseline, greater gains were obtained using a simple two-cluster CAT system [4].[7] One reason for this is that CAT updates all the model parameters in an adaptive framework, whereas only the eigenvoices are updated in ML eigenspace training [26]. The results from Table III indicate that on this task, training all the canonical model parameters, the performance has approximately saturated at about four clusters.[8] The use of a relatively small number of clusters is advantageous when using MPE training. MPE, in common with other discriminative training criteria, is more likely to overtrain than ML training. Thus, the generalization with large numbers of clusters would be expected to be poor.

C. State-of-the-Art Results

From Table III, the eigen-decomposition initialized three-cluster system, without a bias cluster, yielded the best result. However, due to memory limitation, it is only possible to build a two-cluster system for the larger 28-component state-of-the-art[9] configuration. Hence, the two-cluster gender initialization scheme was used here. It should be noted that systems with more clusters may be expected to yield greater gains. For MPE-CAT and MPE-ST, the single-cluster static MPE prior was used. MPE-GD-MAP used more complex prior as indicated in the system description.

Table IV shows a comparison of ML and MPE training. As expected, the use of MPE training to directly generate GD models again gave poor generalization, hence the need for MPE-MAP training of GD models [23]. The GD MPE-MAP and CAT models out-performed the GI models on both test sets. Significance test showed there was a slight gain from using the CAT system over the GD MPE-MAP system for MPE training. This lack of significant difference is not too surprising given the complexity of the initial GI system and the use of a complex scheme to generate a GD MPE-MAP system.

Table V shows the performance of the state-of-the-art systems using unsupervised, transcription mode, test-set adaptation. All adaptively trained systems, MPE-CAT, MPE-SAT, and MPE-ST employed the simplified MPE training scheme. Pairwise significance test showed there was no significant difference between the MPE-CAT system and the GD MPE-MAP system. However, the use of the ST for both training and testing showed significant gains over all the other systems.

[7] Few strict comparisons exist on large vocabulary systems. One close comparison is on the WSJ task where a 20-cluster eigenvoice system was built [26], and a two-cluster CAT system [4] on the SI-284 training data. Despite starting from a better baseline, the CAT system showed a greater relative reduction in WER over the GI system than the equivalent eigenvoice system.

[8] This effect has also been observed for eigenvoices in [27]. where a large number of eigenvoices did not help performance and in some cases degraded the performance.

[9] The GI model set was similar to the model-set used in the CUED RT03 evaluation, though trained on 290 h, rather than 370 h. Given the available training data, this was felt to be best performing system available at CUED. In the evaluation, the complete CUED system achieved the lowest WER on this task.

TABLE IV
28-COMPONENT GI, GD, AND CAT SYSTEM PERFORMANCE USING ML AND MPE TRAINING ON `eval03`

System	ML	MPE
GI	31.5	28.3
GD	31.2	28.5
GD (MPE-MAP)	—	28.0
CAT	31.2	27.8

TABLE V
28-COMPONENT MPE-TRAINED SYSTEMS PERFORMANCE WITH TRAINING AND TEST SET ADAPTATION ON `eval03`

System	Adaptation		WER(%)
	Train	Test	
GI	—	CMLLR	26.1
GD (MPE-MAP)	gender		25.8
SAT	CMLLR		25.9
CAT	CAT	ST	25.7
ST	ST		25.5

VIII. CONCLUSION

This paper has described the application of discriminative training to model sets with multiple clusters. In particular, the application of MPE training to a CAT system has been detailed. However, the form of discriminative adaptive training is applicable to other multiple-cluster systems such as eigenvoices. In order to apply MPE training, modified versions of smoothing functions and parameter priors, I-smoothing, were derived for the multiple-cluster systems.This yielded a minor modification to the statistics required to estimate the MPE CAT canonical model. Practical issues such as the choice of prior and selection of smoothing constant are also discussed. A simple extension to the CAT system, STs, where a CMLLR transform was combined with CAT weights, was also described. The performance of the systems were evaluated on a state-of-the-art conversation telephone speech recognition task. On this task, MPE trained CAT systems significantly out-performed the ML-CAT systems and were slightly better than a state-of-the-art GD MPE-MAP system. Using the more complex STs showed significant gains over standard adaptively trained systems as well as CAT systems. For the state-of-the-art system, it was only possible to use two-clusters. It is hoped that the use of additional clusters will yield greater gains, as illustrated in the smaller development system.

ACKNOWLEDGMENT

The authors would like to thank K. C. Sim and X. Liu, Cambridge University, for helpful discussions. This paper does not necessarily reflect the position or the policy of the U.S. Government and no official endorsement should be inferred.

REFERENCES

[1] T. Anastasakos, J. Mcdonough, R. Schwartz, and J. Makhoul, “A compact model for speaker adaptive training,” in *Proc. Int. Conf. Speech Language Processing*, 1996, pp. 1137–1140.

[2] C. J. Leggetter and P. C. Woodland, “Maximum likelihood linear regression for speaker adaptation of continuous density HMMs,” *Comput. Speech Lang.*, vol. 9, pp. 171–186, 1995.

[3] M. J. F. Gales, “Maximum likelihood linear transformations for HMM-based speech recognition,” *Comput. Speech Lang.*, vol. 12, pp. 75–98, 1998.

[4] ——, “Multiple-cluster adaptive training schemes,” presented at the Int. Conf. Acoustics, Speech, Signal Processing, 2001.

[5] ——, “Cluster adaptive training of hidden Markov models,” *IEEE Trans. Speech Audio Process.*, vol. 8, no. 4, pp. 417–428, Apr. 2000.

[6] R. Kuhn, J. C. Junqua, P. Nguyen, and N. Niedzielski, “Rapid speaker adaptation in eigenvoice space,” *IEEE Trans. Speech Audio Process.*, vol. 8, no. 6, pp. 695–707, Jun. 2000.

[7] L. R. Bahl, P. F. Brown, P. Souza, and R. Mercer, “Maximum mutual information estimation of hidden Markov model parameters for speech recognition,” in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, vol. 1, 1986, pp. 49–52.

[8] B. H. Juang, W. Chou, and C. H. Lee, “Minimum classification error rate methods for speech recognition,” *IEEE Trans. Speech Audio Process.*, vol. 5, no. 2, pp. 266–277, Feb. 1997.

[9] D. Povey and P. C. Woodland, “Minimum phone error and I-smoothing for improved discriminative training,” presented at the Int. Conf. Acoustics, Speech, Signal Processing, Orlando, FL, 2002.

[10] D. Povey, “Discriminative Training for Large Vocabulary Speech Recognition,” Ph.D. dissertation **[AUTHOR: PLEASE PROVIDE DEPT.]**, Cambridge Univ., Cambridge, U.K., 2003.

[11] Y. Normandin, “Hidden Markov Models, Maximum Mutual Information Estimation, and the Speech Recognition Problem,” **[AUTHOR: PLEASE PROVIDE DEPT., CITY, AND PROVINCE]**, McGill Univ., Canada, 1991.

[12] A. Gunawardana and W. Byrne, “Discriminative speaker adaptation with conditional maximum likelihood linear regression,” presented at the EuroSpeech Conf., 2001.

[13] T. S. J. McDonough and A. Waibel, “On maximum mutual information speaker- adapted training,” presented at the Int. Conf. Acoustics, Speech, Signal Processing, Orlando, FL, 2002.

[14] L. Wang and P. C. Woodland, “Discriminative adaptive training using the MPE criterion,” presented at the ASRU Conf., 2003.

[15] K. Yu and M. J. F. Gales, “Discriminative Cluster Adaptive Training,” Eng. Dept., Cambridge Univ., Cambridge, U.K., Tech. Rep. CUED/F-INFENG/TR486, 2004.

[16] ——, “Adaptive training using structured transforms,” presented at the Int. Conf. Acoustics, Speech, Signal Processing, 2004.

[17] M. J. F. Gales, “Cluster adaptive training for speech recognition,” in *Proc. Int. Conf. Speech Language Processing*, 1998, pp. 1783–1786.

[18] X. Huang, A. Acero, and H. Hon, *Spoken Language Processing*. Englewood Cliffs, NJ: Prentice-Hall, 2001.

[19] R. Kuhn, P. Nguyen, J. C. Junqua, L. Goldwasser, N. Niedzielski, S. Fincke, K. Field, and M. Contolini, “Eigenvoices for speaker adaptation,” in *Proc. Int. Conf. Speech Language Processing*, 1998, pp. 1771–1774.

[20] P. Nuguyen, C. Wellekens, and J. C. Junqua, “Maximum likelihood eigenspace and MLLR for speech recognition in noisy environments,” in *Proc. Eurospeech*, 1999, pp. 2519–2522.

[21] J. L. Gauvain and C. H. Lee, “Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains,” in *IEEE Trans. Speech Audio Process.*, vol. 2, Jan. 1994, pp. 291–298.

[22] P. C. Woodland and D. Povey, “Large scale discriminative training of hidden Markov models for speech recognition,” *Comput. Speech Lang.*, vol. 16, pp. 25–48, 2002.

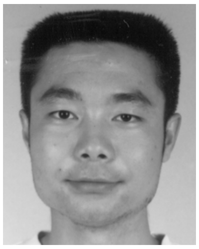
[23] D. Povey, M. J. F. Gales, D. Y. Kim, and P. C. Woodland, “MMI-MAP and MPE-MAP for acoustic model adaptation,” presented at the EuroSpeech Conf., 2003.

[24] M. J. F. Gales, “Acoustic factorization,” presented at the ASRU Conf., 2001.

[25] L. Gillick and S. Cox, “Some statistical issues in the comparison of speech recognition,” in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, 1989, pp. 532–535.

[26] P. Nguyen, R. Kuhn, L. Rigazio, J. C. Junqua, and C. Wellekens, “Self-adaptation using eigenvoices for large-vocabulary continuous speech recognition,” presented at the ITRW Adaptation Conf., 2001.

[27] H. Botterweck, “Very fast adaptation for large vocabulary continuous speech recognition using eigenvoices,” in *Proc. Int. Conf. Speech Language Processing*, 2000, pp. 354–357.



Kai Yu received the B.Eng. and M.Sc. degrees from the Department of Automation, Tsinghua University, China, in 1999 and 2002, respectively.

He is currently a research student in the Machine Intelligencce Laboratory, Engineering Department, Cambridge University, Cambridge, U.K. His research interests include statistical pattern recognition and its application in speech and audio processing. He also worked on the DARPA funded Effective, Affordable and Reusable Speech-to-text (EARS) project from 2002 to 2005.

Mark J. F. Gales (M'XX) **[AUTHOR: IF AN IEEE MEMBER, PLEASE PROVIDE MEMBERSHIP YEAR]** received the B.S. degree in electrical and information sciences and the Ph.D. degree from the University of Cambridge, Cambridge, U.K., in 1988 and 1995, respectively.

He was a Consultant at Roke Manor Research, Ltd. In 1991, he took up a position as a Research Associate in the Speech Vision and Robotics Group, Engineering Department, Cambridge University. From 1995 to 1997, he was a Research Fellow at Emmanuel College, Cambridge. He was then a Research Staff Member with the Speech Group, IBM T. J.Watson Research Center, Yorktown Heights, NY, until 1999, when he returned to the Engineering Department, Cambridge University, as a University Lecturer. He is currently a Reader in information engineering and a Fellow of Emmanuel College.

Dr. Gales was a member of the IEEE Speech Technical Committee from 2001 to 2004.