Automatic Model Complexity Control Using Marginalized Discriminative Growth Functions

Xunying Liu and Mark Gales

Abstract

Selecting the model structure with the "appropriate" complexity, is a standard problem for training large vocabulary continuous speech recognition (LVCSR) systems. State-of-the-art LVCSR systems are highly complex. A wide variety of techniques may be used which alter the system complexity and word error rate (WER). Explicitly evaluating systems for all possible configurations is infeasible, hence an automatic model complexity control criterion is highly desirable. Most existing complexity control schemes can be classified into two types, Bayesian learning techniques and information theory approaches. An implicit assumption is made in both, that increasing the likelihood on held-out data decreases the WER. However this correlation has been found quite weak for current speech recognition systems. This paper presents a novel discriminative complexity control technique, the marginalization of a discriminative growth function. This is a closer approximation to the true WER than standard approaches. Experimental results on a standard LVCSR Switchboard task, showed that marginalized discriminative growth functions outperforms manually tuned systems and conventional complexity control techniques, such as BIC, in terms of WER.

Index Terms

Complexity Control, Discriminative Criteria, Growth Functions

I. Introduction

Selecting the model structure with the "appropriate" complexity is a standard problem for training large vocabulary continuous speech recognition (LVCSR) systems. State-of-the-art LVCSR systems are highly complex. A wide variety of techniques are used which alter both the system complexity and resulting

This work was supported by DARPA grant MDA972-02-1-0013. The paper does not necessarily reflect the position or the policy of the US Government and no official endorsement should be inferred.

word error rate (WER). These techniques include decision tree based state clustering, using a mixture of Gaussians as a state density distribution and various dimensionality reduction schemes. Explicitly evaluating WER for all possible model structural configurations is infeasible. It is therefore useful to find a criterion that predicts the WER ranking order, without explicitly requiring all the systems to be built.

Most existing complexity control schemes can be classified into two types. In *Bayesian learning* techniques model parameters are treated as random variables. The likelihood is integrated out over the model parameters yielding the Bayesian evidence. In the *information theory* approaches a complexity control problem is viewed as finding a minimum code length, for an underlying data generation process [1], [19]. These two approaches are closely related to each other. They asymptotically tend to the Bayesian Information Criterion (BIC) [20] for first order, or Laplace's approximation for second order expansion [26]. These schemes have been previously studied for various complexity control problems for speech recognition systems. For instance, they have been applied to determine the number of states in a decision tree based clustering, or the number of linear transforms for speaker and environment adaptation [22], [23], [2], [21]. An implicit assumption is made in these schemes, that increasing the likelihood on held-out data will decrease the WER. However, this correlation has been found quite weak for current speech recognition systems [10]. It would be preferable to use a complexity control scheme that is more closely related to WER. Along these lines a discriminative measure has previously been used [16] as a method of incrementally splitting Gaussian mixture components. However, no stopping criterion was provided to penalize over-complex model structures.

This paper presents a novel complexity control technique using the marginalization of a *discriminative growth function*, rather than the likelihood in standard Bayesian approaches. Due to sensitivity to outliers, discriminative training criteria, such as maximum mutual information (MMI), cannot be directly integrated over for complexity control. Instead a related discriminative growth function is marginalized. This growth function has the same maxima and minima as the original discriminative criterion, but has a reduced sensitivity to outliers. However, directly marginalizing this discriminative growth function is usually impractical. Using an EM-like approach, a strict lower bound of the growth function can derived. To further reduce computational cost, Laplace's approximation is used for the growth function integration. The discriminative growth function presented in this paper is based on the minimum phone error (MPE) criterion.

In this work HMM systems with mixture of Gaussians as the state output distributions, and multiple heteroscedastic LDA (HLDA) feature transforms are used. HLDA is a linear projection scheme which diagonalizes the feature space [4]. It partitions the entire feature space into a *useful* subspace where

all Gaussian means and variances are kept distinct, and a *nuisance* subspace where component means and variances are globally tied. Multiple HLDA extends this concept so that transforms are shared on a state or Gaussian component level [4], [9]. Thus two forms of system complexity attributes will be determined: the number of components per state; and the number of useful dimensions for each HLDA projection. This problem of simultaneously examining multiple complexity attributes makes commonly used schemes, such as BIC, inappropriate for complexity control [10].

This paper is organized as follows. The next section reviews standard complexity control techniques and their limitations. Section III discusses discriminative training criteria, and issues with the directly marginalization of them for complexity control. One form of discriminative growth function for the MPE criterion is then introduced in section IV. Some implementation issues are discussed in section V. Experimental results on a standard LVCSR task are presented in section VI.

II. Complexity Control

A standard problem in LVCSR training, and machine learning in general, is how to obtain a model structure that generalizes well to unseen data. For speech recognition this generalization is measured usually by WER on unseen data. The aim is to select the optimal model structure $\hat{\mathcal{M}}$ from a set of candidate model structures $\{\mathcal{M}\}$, given a fixed $\mathcal{T}$ length training data set $\mathcal{O} = \{\boldsymbol{o}_1, ..., \boldsymbol{o}_{\mathcal{T}}\}$ and the reference transcription $\mathcal{W}$. An implicit model correctness assumption is made for standard complexity control schemes, hence increasing the unseen data likelihood will decrease the system's WER. In Bayesian techniques, this selection scheme can be expressed as the following *evidence* integral over the model parameters using the training data,

$$\begin{aligned} \hat{\mathcal{M}} &= \arg\max_{\mathcal{M}} P(\mathcal{M}) p(\mathcal{O}|\mathcal{W}, \mathcal{M}) P(\mathcal{W}) \\ &= \arg\max_{\mathcal{M}} P(\mathcal{M}) \int \mathcal{F}_{\mathtt{ml}}(\lambda, \mathcal{M}) p(\lambda|\mathcal{M}) \mathrm{d}\lambda \end{aligned} \tag{1}$$

where λ denotes a parameterization of $\mathcal{M}$. The maximum likelihood (ML) criterion is given by

$$\mathcal{F}_{\mathtt{ml}}(\lambda, \mathcal{M}) = p(\mathcal{O}|\lambda, \mathcal{W}, \mathcal{M}) P(\mathcal{W}). \tag{2}$$

The language model probability $P(\mathcal{W})$ is normally optimized on additional text data. Hence only the structural optimization of acoustic models is considered in this paper. In addition, no prior knowledge over individual model structures is considered, so $P(\mathcal{M})$ is assumed uninformative. The parameter prior

distribution $p(\lambda|\mathcal{M})$ is also treated as uninformative[1].

For HMM based speech recognition systems, it is computationally intractable to directly compute the evidence integral in (1). This has led to a variety of approximation schemes, of which Bayesian Information Criterion (BIC) is the most widely used [20]. BIC may be simply expressed in terms of a penalized log likelihood evaluated at the ML, or *maximum a-posteriori* (MAP), estimate of model parameters $\hat{\lambda}$. The model selection is based on the following approximation

$$\log p(\mathcal{O}|\mathcal{W},\mathcal{M}) \approx \log p(\mathcal{O}|\hat{\lambda},\mathcal{W},\mathcal{M}) - \rho\cdot\frac{k}{2}\log\mathcal{T} \tag{3}$$

where k denotes the number of free parameters in $\mathcal{M}$ and ρ is a penalization coefficient which may be tuned for the specific task [2]. Schwartz proved that when $\rho=1$ BIC is a first order asymptotic expansion of the evidence integral. When the type of model parameters being optimized are very different, for example the number of Gaussians and dimensions, this simple approximation may be poor [10]. The actual form of the parameter, for example whether it is associated with a mean or covariance matrix, can have a large affect on how it alters the likelihood. Thus the simple number of parameters, unless there is a very large amount of data, is a poor measure.

Laplace's approximation provides a second order asymptotic expansion of the evidence integral [26]. The basic idea is to make a local Gaussian approximation of likelihood curvature in the parametric space. The volume under that Gaussian is computed as an approximation to the evidence integral,

$$\begin{aligned}\log p(\mathcal{O}|\mathcal{W},\mathcal{M}) \approx\ & \log p(\mathcal{O}|\hat{\lambda},\mathcal{W},\mathcal{M}) + \frac{k}{2}\log 2\pi \\ & -\frac{1}{2}\log\left|-\nabla^2_{\lambda=\hat{\lambda}}\log p(\mathcal{O}|\lambda,\mathcal{W},\mathcal{M})\right|\end{aligned} \tag{4}$$

$|\cdot|$ denotes the determinant of a matrix, and again $\hat{\lambda}$ is the optimal model parameters. This second order approximation allows the difference in the form of parameters being optimized to be better accounted for.

One issue with both of the above schemes is that the log-likelihood for each model structure is required. For HMMs this can be computationally expensive. One method to avoid this is to derive a lower bound that may be assumed to be applicable for multiple different structures. Let $\tilde{\lambda}$ denote the

[1]Alternatively assumptions may be made about the prior distribution's structure, which typically constrain it to be a conjugate prior distribution for $p(\mathcal{O}|\lambda,\mathcal{W},\mathcal{M})$, so that the evidence integration problem may be relatively simplified. Nevertheless how to select a form of the parameter prior distribution is always subjective.

current parameterization for $\mathcal{M}$. Using a standard EM approach the following inequality may be derived

$$\begin{aligned}\log p(\mathcal{O}|\lambda,\mathcal{W},\mathcal{M}) &\geq \mathcal{L}_{\mathtt{ml}}(\lambda,\tilde{\lambda}) \\ &= \log p(\mathcal{O}|\tilde{\lambda},\mathcal{W},\mathcal{M}) + \mathcal{Q}_{\mathtt{ml}}(\lambda,\tilde{\lambda}) - \mathcal{Q}_{\mathtt{ml}}(\tilde{\lambda},\tilde{\lambda}) \end{aligned} \quad (5)$$

where the auxiliary function, $\mathcal{Q}_{\mathtt{ml}}(\lambda,\tilde{\lambda})$, is given by

$$\mathcal{Q}_{\mathtt{ml}}(\lambda,\tilde{\lambda}) = \sum_{j,\tau} \gamma_j(\tau) \log p(\boldsymbol{o}_\tau|\theta_\tau = s_j,\lambda,\mathcal{M}) \quad (6)$$

and $\theta_\tau = s_j$ indicates that an acoustic observation $\mathbf{o}_\tau$ was generated by state s_j, and the state posterior $\gamma_j(\tau) = P(\theta_\tau = s_j|\mathcal{O},\mathcal{W},\tilde{\lambda},\mathcal{M})$. For LVCSR training the majority of the time is spent accumulating these sufficient statistics to estimate the model parameters. Thus, accumulating these statistics for all possible systems is infeasible. To handle this problem, a range of model structures are required to use information derived from the same set of statistics generated using a single system. For example when determining the number of components, statistics for systems with fewer components per state can be derived by merging appropriate statistics together from a more complex system. This allows the lower bound in (5) to be efficiently computed. This lower bound for the evidence is then used

$$p(\mathcal{O}|\mathcal{W},\mathcal{M}) \geq \int \exp\left(\mathcal{L}_{\mathtt{ml}}(\lambda,\tilde{\lambda})\right) p(\lambda|\mathcal{M})\mathrm{d}\lambda \quad (7)$$

The only term in the lower bound in (5) which is dependent on the model parameters, λ, is the auxiliary function $\mathcal{Q}_{\mathtt{ml}}(\lambda,\tilde{\lambda})$. When multiple model structures use the same set of statistics, the rank ordering derived from the marginalization of the lower bound, $\mathcal{L}_{\mathtt{ml}}(\lambda,\tilde{\lambda})$ is equivalent to that of $\mathcal{Q}_{\mathtt{ml}}(\lambda,\tilde{\lambda})$. However, when multiple sets of statistics are used, the other terms in the lower bound cannot be ignored and must be computed. To further reduce the computational cost, the right hand side of the inequality in (7) can be efficiently approximated using the BIC, or Laplace's approximation.

This form of approximation is related to another lower bound approach, variational approximations [5], [25]. Variational methods yield an alternative form of lower bound, which have been used for complexity control [25] and may also be used with the discriminative approaches discussed in this paper. Markov Chain Monte Carlo (MCMC) sampling schemes are another approach to approximate the evidence integral [14]. In practice MCMC approaches are not generally practical for LVCSR tasks given the high dimensionality of the sampling space.

III. Discriminative Training Criteria

ML training, and hence the related complexity control schemes, yield the best classifier under a number of assumptions, including that the acoustic model is a correct "generative" model for speech. However, for

current speech recognition systems HMMs are not the correct generative models. Discriminative training does not make this assumption, but rather tries to reduce the expected classification error directly. Most state-of-the-art LVCSR systems are now trained using discriminative approaches [3].

A. Maximum Mutual Information (MMI)

One widely used discriminative training criterion is maximum mutual information (MMI) [27], [18]. This is equivalent to maximizing the *a-posteriori* probability of the correct transcription $\mathcal{W}$, given the training data and model.

$$\mathcal{F}_{\texttt{mmi}}(\lambda, \mathcal{M}) = P(\mathcal{W}|\mathcal{O}, \lambda, \mathcal{M}) = \frac{p(\mathcal{O}, \mathcal{W}|\lambda, \mathcal{M})}{p(\mathcal{O}|\lambda, \mathcal{M})} \tag{8}$$

The denominator term, $p(\mathcal{O}|\lambda, \mathcal{M})$, is obtained using a "composite" model by summing over all possible hypotheses $\{\tilde{\mathcal{W}}\}$, including the correct word sequence [2]. When the language model parameters associated with $P(\mathcal{W})$ are fixed during training, the MMI criterion is equivalent to the conditional maximum likelihood (CML) training [13][3]. Though MMI has been successfully used for training LVCSR systems, it has been observed that the MMI criterion gives undue weights to outliers. Utterances with very low posterior probability of the correct transcription can dominate the criterion computation [24].

B. Minimum Phone Error (MPE)

MMI is based on the sentence posterior. However, in speech recognition the most commonly used performance measurement is WER. Minimum word error (MWE) is a criterion more closely related to word, rather than sentence, error rates. It uses a continuous form of WER approximation [8], [6] and the accuracy contribution from each hypothesis is weighted by its posterior probability. A closely related criterion is minimum phone error (MPE). Instead of evaluating recognition accuracy at a word level, a phone level accuracy is calculated under the constraint of the reference word transcription [17], [18]. MPE has been found to achieve better generalisation than MWE [18]. The MPE criterion is expressed

[2]In practice a lattice or N-best list with a finite number of alternatives is used. The language model probability is scaled by a constant $\alpha > 0$ to compensate for the dynamic range issue.

$$p(\mathcal{O}|\lambda, \mathcal{M}) \quad = \quad \sum_{\tilde{\mathcal{W}}} p(\mathcal{O}|\tilde{\mathcal{W}}, \lambda, \mathcal{M}) P(\tilde{\mathcal{W}})^{\alpha}$$

[3]In most research on MMI training, only acoustic model parameters are optimized .

as the average phone correctness of all possible word sequences $\{\tilde{\mathcal{W}}\}$,

$$\mathcal{F}_{\texttt{mpe}}(\lambda,\mathcal{M}) \quad = \quad \sum_{\tilde{\mathcal{W}}} \frac{p(\mathcal{O},\tilde{\mathcal{W}}|\lambda,\mathcal{M})\mathcal{A}(\tilde{\mathcal{W}},\mathcal{W})}{p(\mathcal{O}|\lambda,\mathcal{M})} \tag{9}$$

where $\mathcal{A}(\tilde{\mathcal{W}},\mathcal{W})$ is the accuracy measure of a word sequence $\tilde{\mathcal{W}}$, compared with the reference transcription $\mathcal{W}$. This calculation would require a dynamic programming procedure. However an efficient phone accuracy approximation for lattices was proposed in [17], [18]. First the phone level accuracy for each arc in the lattice against the reference transcript is computed, recognition errors caused by either substitution, deletion or insertion are accounted for. Then the accuracy measure of each arc is smoothed using a forward-backward algorithm like procedure. This acts to de-weight the accuracy of lattice arcs which have very low posterior probabilities, and scale up the accuracy of those that are more likely. A more detailed description of the algorithm is given in [18].

C. Marginalizing Discriminative Criteria

Discriminative complexity control could be achieved by directly replacing the ML criterion in the evidence integral of (1) by a discriminative criterion. If the MPE criterion is used and the model prior, $P(\mathcal{M})$, is assumed informative, this yields

$$\hat{\mathcal{M}} \quad = \quad \arg\max_{\mathcal{M}} \int \mathcal{F}_{\texttt{mpe}}(\lambda,\mathcal{M}) p(\lambda|\mathcal{M}) \mathrm{d}\lambda \tag{10}$$

A similar form of integral of MMI criterion may also be considered for complexity control. However, for both criteria, such a direct marginalization may be inappropriate. The primary reason is that undue weight is given to outliers. State-of-the-art large vocabulary speech recognition systems are trained with hundreds of hours of data and outliers, which are far from the decision boundary, are likely to exist. These outliers are often utterances with very low likelihood, and associated high recognition error rate. This sensitivity to outliers is a well known feature of the MMI criterion [24]. Sentences with very low posteriors are heavily weighted. The performance ranking prediction will be distorted due to the presence of these outliers. The same issue exists with the MPE criterion for sentences with very high recognition error rate.

IV. Discriminative Growth Functions

The sensitivity to outliers mentioned in the previous section could be addressed by explicitly de-weight the outliers utterances [24]. However, in this paper an alternative approach based on *discriminative growth functions* is used. This growth function maintains some of the attributes of the original discriminative

criterion, but is less sensitive to outliers. The marginalization of this growth function is used to determine the appropriate model complexity. This method is very different to the standard Bayesian techniques. As discussed in section II, techniques like BIC and variational method are both approximation schemes to the Bayesian evidence, the marginalization of the ML criterion. In contrast the method proposed here is the marginalization of a discriminative measure, or "discriminative evidence". In this section a general form of growth functions for discriminative criteria is introduced. Then an appropriate form of discriminative growth function is derived for the MPE criterion.

A. General Form of Growth Functions

The form of growth function considered here is applicable to any discriminative criterion which can be expressed as a ratio between two polynomials with positive coeffi cients. The MMI and MPE criteria are in this category. Consider a discriminative training criterion expressed in the following form (the model structure $\mathcal{M}$ is omitted for clarity in this section)

$$\mathcal{F}(\lambda) \quad = \quad \frac{\mathcal{F}_{\mathtt{num}}(\lambda)}{\mathcal{F}_{\mathtt{den}}(\lambda)} \tag{11}$$

The general form of a growth function proposed here can be expressed as,

$$\mathcal{G}(\lambda) \quad = \quad \mathcal{F}_{\mathtt{den}}(\lambda) \left[\mathcal{F}(\lambda) - \mathcal{F}(\tilde{\lambda}) + C\mathcal{F}_{\mathtt{sm}}(\lambda, \tilde{\lambda}) \right] \tag{12}$$

where $\tilde{\lambda}$ is the *current* parameter estimate. The fi rst two terms in the bracket give information about the curvature of the criterion surface in the parametric space. The third regularization term in the bracket is a smoothing term, scaled by a positive constant, C. As explained in section III-C, for speech recognition systems outliers may exist in the training data. To reduce the growth function's sensitivity to outliers, the smoothing criterion should be selected to compensate for the low likelihood, or high error rate, contribution from these outliers. Thus the smoothing term will be associated with the likelihood or WER. The constant C in (12) determines the effect from this smoothing criterion. The exact form of $\mathcal{F}_{\mathtt{sm}}(\lambda, \tilde{\lambda})$, depends on the underlying discriminative criterion being considered, and will be further discussed in the next section for the MPE criterion. In addition, the denominator term, $\mathcal{F}_{\mathtt{den}}(\lambda)$, outside the bracket in (12), may also help to reduce the sensitivity to outliers. This is the case for both the MMI and MPE criteria where it is associated with the likelihood of a sentence, $\mathcal{F}_{\mathtt{den}}(\lambda) = p(\mathcal{O}|\lambda)$. Hence highly unlikely word sequences will have a smaller effect on the growth function.

The gradient of the growth function, $\mathcal{G}(\lambda)$, may be expressed as

$$\begin{aligned}\frac{\partial \mathcal{G}(\lambda)}{\partial \lambda} &= \left[\mathcal{F}(\lambda) - \mathcal{F}(\tilde{\lambda}) + C\mathcal{F}_{\mathtt{sm}}(\lambda,\tilde{\lambda})\right]\frac{\partial \mathcal{F}_{\mathtt{den}}(\lambda)}{\partial \lambda} \\ &\quad + \mathcal{F}_{\mathtt{den}}(\lambda)\left[\frac{\partial \mathcal{F}(\lambda)}{\partial \lambda} + C\frac{\partial \mathcal{F}_{\mathtt{sm}}(\lambda,\tilde{\lambda})}{\partial \lambda}\right] \end{aligned} \tag{13}$$

Hence a turning point of the original criterion is also a turning point of the growth function around the current parameter estimate $\tilde{\lambda}$ when C approaches zero. This constrains the attributes of the growth function to be related to those of the original criterion.

B. MPE Growth Function

This paper focuses on using a growth function based on the MPE criterion. The growth function considered is

$$\mathcal{G}(\lambda) = p(\mathcal{O}|\lambda)\left[\mathcal{F}_{\mathtt{mpe}}(\lambda) - \mathcal{F}_{\mathtt{mpe}}(\tilde{\lambda}) + C\mathcal{F}_{\mathtt{sm}}(\lambda,\tilde{\lambda})\right] \tag{14}$$

where the smoothing term is given by

$$\mathcal{F}_{\mathtt{sm}}(\lambda,\tilde{\lambda}) = \sum_{\substack{\tilde{\mathcal{W}} \\ \mathcal{A}(\tilde{\mathcal{W}},\mathcal{W})<\mathcal{F}_{\mathtt{mpe}}(\tilde{\lambda})}} P(\tilde{\mathcal{W}}|\mathcal{O},\lambda)\left[\mathcal{F}_{\mathtt{mpe}}(\tilde{\lambda}) - \mathcal{A}(\tilde{\mathcal{W}},\mathcal{W})\right] \tag{15}$$

This smoothing criterion has the attributes discussed in section III-C, as the effect of word sequences whose accuracy are below the average level is reduced. In addition the term outside the bracket in the MPE growth function, $p(\mathcal{O}|\lambda)$, is associated with the likelihood of a sentence and will further reduce the sensitivity to outliers.

Direct marginalization of the growth function in (14) is difficult for speech systems, due to the latent variables associated with HMMs, and will be highly inefficient for complexity control. An approach similar to that discussed in section II is therefore used. A lower bound for the MPE growth function may be derived [12],

$$\mathcal{L}_{\mathtt{mpe}}(\lambda,\tilde{\lambda}) = \log \mathcal{G}(\tilde{\lambda}) + \frac{\mathcal{Q}_{\mathtt{mpe}}(\lambda,\tilde{\lambda}) - \mathcal{Q}_{\mathtt{mpe}}(\tilde{\lambda},\tilde{\lambda})}{\sum_{j,\tau}\gamma_j^{\mathtt{mpe}}(\tau)} \tag{16}$$

where the MPE "auxiliary function" is given by [4]

$$\mathcal{Q}_{\mathtt{mpe}}(\lambda,\tilde{\lambda}) = \sum_{j,\tau}\gamma_j^{\mathtt{mpe}}(\tau)\log p(\boldsymbol{o}_\tau|\theta_\tau = s_j,\lambda) \tag{17}$$

[4]Only the optimization of Gaussian means and variances are considered.

and $\gamma_j^{\mathtt{mpe}}(\tau)$ is the MPE hidden state occupancy. The following lower bound marginalization is then used for complexity control.

$$\hat{\mathcal{M}} \;=\; \arg\max_{\mathcal{M}} \int \exp\left(\mathcal{L}_{\mathtt{mpe}}(\lambda,\tilde{\lambda})\right) p(\lambda|\mathcal{M})\mathrm{d}\lambda \tag{18}$$

Finally, although the dependency upon latent variables has been removed for the growth function lower bound, the marginalization in (18) is still non-trivial. To solve this problem, the integral in (18) may be computed using Laplace's approximation, in common with the ML bound marginalization in (5).

The calculation of the growth function lower bound requires the MPE occupancy statistics $\left\{\gamma_j^{\mathtt{mpe}}(\tau)\right\}$. For the MPE growth function, the hidden state occupancy $\gamma_j^{\mathtt{mpe}}(\tau)$ in (17) is given by [12]

$$\begin{aligned}\gamma_j^{\mathtt{mpe}}(\tau) \;&=\; \gamma_j^{\mathtt{num}}(\tau) - \gamma_j^{\mathtt{den}}(\tau) \\ &\quad -C \sum_{\tilde{\mathcal{W}},\gamma_{\tilde{\mathcal{W}}}^{\mathtt{mpe}}<0} P(\theta_\tau = s_j|\mathcal{O},\tilde{\mathcal{W}},\tilde{\lambda})\gamma_{\tilde{\mathcal{W}}}^{\mathtt{mpe}}\end{aligned} \tag{19}$$

where the MPE word sequence occupancy is defined as

$$\gamma_{\tilde{\mathcal{W}}}^{\mathtt{mpe}} = P(\tilde{\mathcal{W}}|\mathcal{O},\tilde{\lambda})\left[\mathcal{A}(\tilde{\mathcal{W}},\mathcal{W}) - \mathcal{F}_{\mathtt{mpe}}(\tilde{\lambda})\right] \tag{20}$$

and the numerator and denominator occupancies are given by

$$\begin{aligned}\gamma_j^{\mathtt{num}}(\tau) &= \sum_{\tilde{\mathcal{W}}} P(\theta_\tau = s_j|\mathcal{O},\tilde{\mathcal{W}},\tilde{\lambda})\gamma_{\tilde{\mathcal{W}}}^{\mathtt{mpe}} \quad (\gamma_{\tilde{\mathcal{W}}}^{\mathtt{mpe}} \geq 0) \\ \gamma_j^{\mathtt{den}}(\tau) &= -\sum_{\tilde{\mathcal{W}}} P(\theta_\tau = s_j|\mathcal{O},\tilde{\mathcal{W}},\tilde{\lambda})\gamma_{\tilde{\mathcal{W}}}^{\mathtt{mpe}} \;(\gamma_{\tilde{\mathcal{W}}}^{\mathtt{mpe}} < 0)\end{aligned} \tag{21}$$

It is interesting to compare the MPE occupancy derived from the growth function, given in (19), with the standard form used in LVCSR MPE training [18] given by

$$\begin{aligned}\gamma_j^{\mathtt{mpe}}(\tau) \;&=\; \gamma_j^{\mathtt{num}}(\tau) - \gamma_j^{\mathtt{den}}(\tau) \\ &\quad -E \sum_{\tilde{\mathcal{W}},\gamma_{\tilde{\mathcal{W}}}^{\mathtt{mpe}}<0} P(\theta_\tau = s_j|\mathcal{O},\tilde{\mathcal{W}},\tilde{\lambda})\gamma_{\tilde{\mathcal{W}}}^{\mathtt{mpe}}\end{aligned} \tag{22}$$

where a constant $E > 0$ is empirically tuned. These two forms of MPE occupancy are equivalent to one another when $E = C$. However the two smoothing terms serve very different purposes. The smoothing term in the standard MPE occupancy, in (22), ensures a stable convergence during training. Whereas, the smoothing term derived from the growth function helps reduce the sensitivity to outliers sentences with high error rates.

The growth function lower bound in (16) has a similar form to the log-likelihood bound in (5). Both may be expressed as the value of the underlying objective function at the current parameter estimate,

$\tilde{\lambda}$, plus a second term that is related to the difference in auxiliary functions. In the same fashion as the log-likelihood bound, for efficiency multiple complexity configurations should make use of a single set of statistics. In this case, the only term that will determine the rank-ordering of the systems will be the MPE auxiliary function, $\mathcal{Q}_{\texttt{mpe}}(\lambda, \tilde{\lambda})$. The form of statistic merging used in this work is discussed in more detail in the next section. One important aspect for both this discriminative bound and the log-likelihood bound is the accuracy of the derived statistics. As the differences between the structure used to derive the statistics and the model being considered increases, the bound may become increasingly looser and the performance ranking poorer. To reduce this effect a maximum structural mutation from the system used to derive the statistics may be enforced. This will also be discussed in more detail in the next section.

Another issue with using growth functions for complexity control is the setting of the regularization constant C. The setting of this constant has two effects. First, it controls the contribution from the smoothing term of the MPE occupancy, given in (19), to reduce the sensitivity to outliers. Second, the setting of C may affect the selection of the optimal configuration, and the speed of structural mutation from the current model. This constant may need to be appropriately set for complexity control. However, it is found in practice that the WER performance of the final system is not sensitive to the setting of C, when it is sufficiently large. Taking the MPE GFunc system on the 297 hour training set in table I of section VI as an example. Using all the configurations described in sections V and VI, a total of seven MPE GFunc systems were built using varying C values in the set $\{0.0, 0.5, 1.0, 1.5, 2.0, 3.0, 4.0\}$. This is also the range used to investigate the effect of C as a smoothing constant on standard discriminative training in previous research [18], [27]. First, it was found that the WER performances were fairly robust against the setting of C, varying between 33.9% ($C = 0$ and $C = 0.5$) and 33.8% for all other five GFunc systems. This shows that the performances of MPE GFunc systems are not sensitive to the setting of the smoothing constant C. This is an important feature of a good complexity control technique as no parameters requires excessive tuning. Second, the relationship between the average number of Gaussians per state in the final system and C was examined. This is shown in Fig. 1. As expected, increasing the value of C, when it is sufficiently large, for example when $C > 1.0$, led to increased model complexity in the final system. This is simply a natural aspect of the smoothing term, as it is increasingly dominating the growth function calculation. The same property of C also holds during standard discriminative training as it controls the speed and stability of the criterion optimization [18], [27], [15]. Another interesting observation is that an over-small valued C was found to cause unstable structural mutation. For example, when $C = 0$, a highly complex model structure with more than 18.4 components per state was selected. This most complex system also gave the poorest WER performance, 33.9%. Once again this may attribute

to the nature of C as a smoothing constant. Due to the above reasons, for all the experiments in this paper the value of C was set to 2.0 and not altered. This is also a standard value used for MPE training.

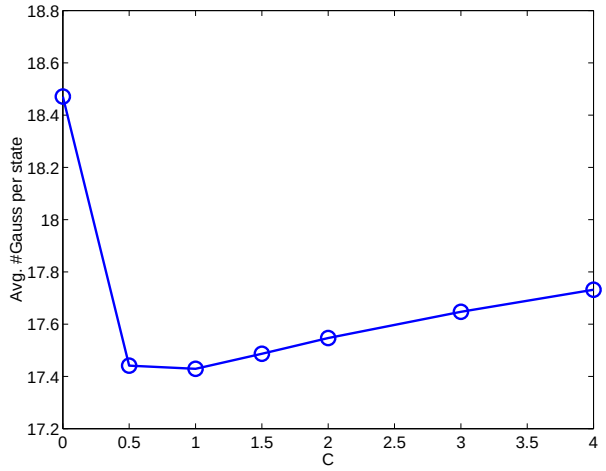

Fig. 1. VARYING C VS. AVG. #GAUSS PER STATE FOR A GLOBAL TRANSFORM HLDA SYSTEM ON 297 HOUR `h5etrain03`

V. IMPLEMENTATION ISSUES

In this section implementation issues for using marginalized discriminative growth functions will be discussed.

A. Statistic Merging

As discussed in section IV-B, the same set of statistics may be used for a range of model structures. As it is only possible to merge statistics, the number of components, or other complexity control attribute, can only be reduced. For this merging process, the statistics from a pair of Gaussians must be combined to form a single Gaussian. This is a standard problem and is solved by simply combining the appropriate first, or second, order statistics and the occupancy counts. For example when joining component j and k to yield l, and using the MPE statistics as an example, this yields

$$\gamma_l^{\mathtt{mpe}}(\tau) \quad = \quad \gamma_j^{\mathtt{mpe}}(\tau) + \gamma_k^{\mathtt{mpe}}(\tau) \tag{23}$$

and similarly for the first and second order statistics. The same merging will also be performed for the ML statistics for the log-likelihood lower-bound. All possible pairs of component merging are considered. The one with the largest increase in the objective function is selected.

B. Constrained Maximum Structural Mutation

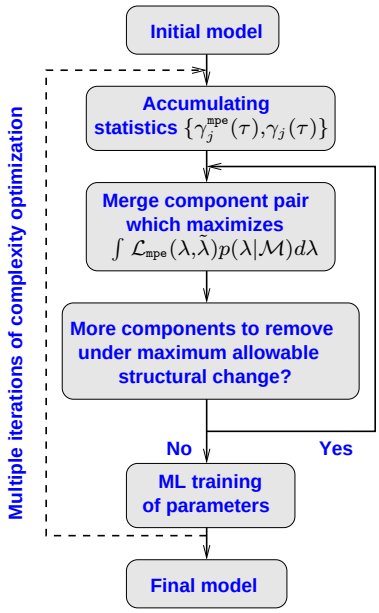

Fig. 2. SELECTING THE NUMBER OF GAUSSIAN COMPONENTS PER STATE USING MARGINALIZED MPE GROWTH FUNCTIONS VIA COMPONENT MERGING

For effi ciency, the lower bound of the discriminative growth function, or log-likelihood, is derived from statistics of a single system as discussed above. As the structural mutation from the current model is increased, the reliability of the fi xed statistics will decrease. This may lead to a poor selection of model complexity. To overcome this problem, the whole structural optimization process can be performed in an iterative mode. An overview of the algorithm, when using marginalized MPE growth functions to select the number of Gaussians per state, is shown in Fig. 2. A maximum mutation in the model complexity is imposed. For instance, the maximum number of Gaussians that may be removed from any state per iteration is constrained. In the work here the maximum mutation was set to be 2. Between iterations of structural optimization, model parameters were re-estimated using ML training to obtain improved statistics.

By replacing the growth function integral in the fi gure with BIC, and using the log-likelihood bound in (5), the same procedure was also applied to BIC. In all experiments a total of four iterations of complexity control were performed.

C. Hessian Approximation

The lower bound marginalization for the MPE growth function in (16), may be approximated via Laplace's approximation. However this requires the storage of a Hessian matrix with respect to all the model parameters. However, the number of model parameters in an LVCSR system can be in the millions making the storage and calculation of the Hessian impractical. To solve this problem, assumptions can be made about the structure of the Hessian. In particular, by assuming that the Hessian has a block diagonal structure [10], [9] the problem is tractable. The exact form of the approximated Hessian depends on that of the growth function lower bound. Let $\check{\boldsymbol{o}}_\tau^{(r_j)} = \boldsymbol{A}^{(r_j)}\boldsymbol{o}_\tau$ denote the projected feature after the HLDA transform, where $\boldsymbol{A}^{(r_j)}$ denotes the HLDA transform that component j is assigned to. Let $\check{\boldsymbol{\mu}}^{(j)}$, $\check{\boldsymbol{\Sigma}}^{(j)}$ denote the component means and covariances in the transformed space. The MPE auxiliary function in (17) may can be expressed as

$$\begin{aligned}\mathcal{Q}_{\texttt{mpe}}(\lambda,\tilde{\lambda}) &= \frac{1}{2}\sum_{j,\tau}\gamma_j^{\texttt{mpe}}(\tau)\left\{\log\left|\boldsymbol{A}^{(r_j)}\right|^2 - \log\left|\check{\boldsymbol{\Sigma}}^{(j)}\right|\right. \\ &\left. -(\check{\boldsymbol{o}}_\tau^{(r_j)} - \check{\boldsymbol{\mu}}^{(j)})^\top\check{\boldsymbol{\Sigma}}^{(j)-1}(\check{\boldsymbol{o}}_\tau^{(r_j)} - \check{\boldsymbol{\mu}}^{(j)})\right\} \end{aligned} \quad (24)$$

Each Gaussian component is assumed to be independent of all others. Furthermore within each Gaussian component, the mean, variance and each row of the HLDA transforms are also assumed independent of each other. For the integral over the growth function's lower bound in (18), the log-determinant of the Hessian matrix may be approximated as

$$\begin{aligned}\log\left|-\nabla_\lambda^2\mathcal{Q}_{\texttt{mpe}}(\lambda,\tilde{\lambda})\right| &\approx \sum_{r,i}\log\left|-\frac{\partial^2\mathcal{Q}_{\texttt{mpe}}(\lambda,\tilde{\lambda})}{\partial^2\boldsymbol{a}_i^{(r)}}\right| \\ +\sum_j\log\left|-\frac{\partial^2\mathcal{Q}_{\texttt{mpe}}(\lambda,\tilde{\lambda})}{\partial^2\check{\boldsymbol{\mu}}^{(j)}}\right| &+\sum_j\log\left|-\frac{\partial^2\mathcal{Q}_{\texttt{mpe}}(\lambda,\tilde{\lambda})}{\partial^2\check{\boldsymbol{\Sigma}}^{(j)}}\right| \end{aligned} \quad (25)$$

where the second order differentials are derived from (24) and yield

$$\begin{aligned}\frac{\partial^2\mathcal{Q}_{\texttt{mpe}}(\lambda,\tilde{\lambda})}{\partial^2\check{\boldsymbol{\mu}}^{(j)}} &= -\frac{1}{2}\sum_\tau\gamma_j^{\texttt{mpe}}(\tau)\check{\boldsymbol{\Sigma}}^{(j)-1} \\ \frac{\partial^2\mathcal{Q}_{\texttt{mpe}}(\lambda,\tilde{\lambda})}{\partial^2\check{\boldsymbol{\Sigma}}^{(j)}} &= -\frac{1}{2}\sum_\tau\gamma_j^{\texttt{mpe}}(\tau)\left[2(\check{\boldsymbol{o}}_\tau^{(r_j)} - \check{\boldsymbol{\mu}}^{(j)})\right. \\ &\left.\times(\check{\boldsymbol{o}}_\tau^{(r_j)} - \check{\boldsymbol{\mu}}^{(j)})^\top\check{\boldsymbol{\Sigma}}^{(j)-3} - \check{\boldsymbol{\Sigma}}^{(j)-2}\right] \\ \frac{\partial^2\mathcal{Q}_{\texttt{mpe}}(\lambda,\tilde{\lambda})}{\partial^2\boldsymbol{a}_i^{(r)}} &= -\frac{\boldsymbol{c}_i^{(r)}\boldsymbol{c}_i^{(r)\top}}{\left|\boldsymbol{A}^{(r)}\right|^2}\sum_{j\in r,\tau}\gamma_j^{\texttt{mpe}}(\tau) - \boldsymbol{G}^{(r,i)} \end{aligned} \quad (26)$$

where $\boldsymbol{c}_i^{(r)}$ denotes the cofactor vector associated with row $\boldsymbol{a}_i^{(r)}$ and the transform specific statistics $\left\{\boldsymbol{G}^{(r,i)}\right\}$ are accumulated on a row by row basis

$$\boldsymbol{G}^{(r,i)} = \sum_{j \in r,\tau} \frac{\gamma_j^{\mathtt{mpe}}(\tau)}{\check{\sigma}_i^{(j)2}} (\boldsymbol{o}_\tau - \boldsymbol{\mu}^{(j)})(\boldsymbol{o}_\tau - \boldsymbol{\mu}^{(j)})^\top \tag{27}$$

As $\boldsymbol{G}^{(r,i)}$ is accumulated using statistics from the original feature-space, there is no need to perform statistic merging as described in section V-A for multiple Gaussian components. The same statistics can be used to generate a range of sizes of useful dimension. Note, that this assumes that the assignment of component to transform is fixed, which is the situation considered in this paper.

VI. Results

In this section the performance of the complexity control schemes are evaluated on a large vocabulary conversational telephone speech (CTS) task. Three CTS training configurations were used. The first is a full system using a 297 hour training set `h5etrain03`, consisting of 4800 Switchboard I, 228 Call Home English (CHE) and 418 Linguistic Data Consortium (LDC) Cellular conversation sides [3]. Two subsets of this were also used: 46 hour `minitrain04`; 76 hour `h5etrain03sub`. Both subsets were selected to have the same gender and channel condition distribution of the full set. The total number of training speakers in the full set is approximately 8 times as the 46 hour subset and 4 times the 76 hour. Note the 46 hour `minitrain04` is subsumed by `h5etrain03sub`. For evaluation a 3 hour subset of 2001 development data, `dev01sub`, was used. The test set contains 20 Switchboard I and 20 Switchboard II phase II conversation sides of the NIST LVCSR evaluation data in 2000 and 1998 respectively, and another 19 LDC Cellular sides. The data was parameterized using PLP Cepstral features normalized on conversation side basis, using vocal tract length normalization (VTLN). The data was further normalized using Cepstral mean and variance normalization. The baseline feature vector used for all projections was a 52 dimensional acoustic feature generated by appending derivatives up to the third order. For the baseline configuration this 52-dimensional feature vector was projected down to 39 dimensions using HLDA. For all these multiple HLDA systems the silence Gaussians were assigned to one transform class, while the speech Gaussians were split into 64 distinct classes. The component assignment used a top-down splitting procedure, based on distance measure of Gaussian components in the acoustic space. Continuous density, mixture of Gaussians, cross-word triphone, gender independent HMM systems were used. After phonetic decision tree based tying, there are approximately 3k speech states for the 46 hour subset, and 6k states for the other two sets. Unless otherwise stated ML training were used for all systems. The lattices used to obtain the MPE statistics for both the complexity control experiments and MPE training were generated

using the baseline 39-dimensional global HLDA system. All recognition experiments used a 58k word trigram language model.

Three forms of complexity control were compared to the baseline approach of using a fixed number of components for each state in the system, referred to as *Fixed*. *VarMix* [3] is a simple approach where the number of components in a state is set to be proportional to the number of frames assigned to that state raised to a power, in these experiments 0.2. The total number of components in the system is fixed so that the average number of Gaussians per state remains the same as standard, Fixed, system from which it was derived[5]. *BIC*, as an example of a Bayesian complexity control, was implemented using the efficient approach discussed in section V-B. Finally the MPE discriminative evidence framework, *MPE GFunc*, was evaluated.

Initially only the number of components associated with each state was determined. The front-end for these experiments used the standard global HLDA projection to 39 dimensions. Table I shows the performance of the baseline system for a range of fixed number of components per state. Using VarMix complexity control gains of 0.1%-0.3% absolute over the baseline Fixed system were obtained. This is not surprising as the amount of data associated with each state can vary considerably. The best VarMix results were all associated with the most complex configuration using 20 Gaussian components per state. The results using BIC, along with the average number of components per state, is also shown in Table I. One interesting issue with the iterative complexity control used here for both the BIC and the GFunc systems is the selection of the initial model. This starting model may affect both the complexity and WER of the final system, as it is not possible to have a final system that is more complex using the approach in section V-B. Hence it is preferable to used a starting model that has a lower error rate, and possibly a more compact model structure. In Table I the starting models used to obtain the initial statistics and determine the maximum complexity of the systems are marked with a "$\star$". These are selected based on the WER performances of the baseline Fixed systems. The three systems were 20 components for the 46 hour subset, 16 components for 76 hour and 20 components for the 297 hour systems. Note that for the 76 hour set, the baseline system had more than twice the number of tied states than the 46 hour set. Three values of ρ in (3) were used, 0.5, 1.0 and 2.0. Note the baseline system marked with a "$\star$" is the system with $\rho = 0$. Though for all data sets the performance of the best BIC system was better than that of the baseline system, it was only comparable to the best VarMix system, but with fewer components per state. For example, on the 76 hour set, the best BIC system ($\rho = 0.5$), which had 15.57

[5]Strictly this is not a complexity control approach as the total number of components is not determined.

Complexity Control		WER%		
		46 hr	76 hr	297 hr
Fixed	12	38.3	36.1	35.1
	14	38.0	36.0	34.8
	16	37.8	$35.8^{\star}$	34.9
	18	37.9	35.8	34.3
	20	$37.8^{\star}$	35.8	$34.1^{\star}$
VarMix Avg #Gauss	12	37.9	36.1	34.9
	14	37.7	35.8	34.7
	16	37.6	35.7	34.3
	18	37.6	35.7	34.0
	20	37.5	35.6	33.9
BIC (ρ = 0.5)		37.4	35.7	34.1
(#Gauss)		(19.38)	(15.57)	(19.21)
BIC (ρ = 1.0)		37.4	35.8	34.2
(#Gauss)		(18.45)	(14.68)	(18.68)
BIC (ρ = 2.0)		37.5	36.1	34.2
(#Gauss)		(18.04)	(12.73)	(17.71)
MPE GFunc		37.2	35.7	33.8
(#Gauss)		(18.34)	(14.52)	(17.54)

TABLE I

OPTIMIZING #GAUSS FOR GLOBAL TRANSFORM HLDA SYSTEMS, $\star$ INDICATES THE INITIAL MODEL FOR BIC AND MPE GFUNC SYSTEMS.

components per state on average, matched the performance of the more complex 18 component VarMix system. The performance using the marginalized MPE growth are also shown in Table I. In contrast to the VarMix or BIC approach there was no tuning of any free parameters. For all of the three sets, the MPE GFunc system outperformed, or approximately matched, the best BIC and VarMix systems with fewer parameters. This indicates that the MPE GFunc system is able to select configurations that make more efficient use of the number of components.

To further investigate marginalized growth functions for model selection, a more complex problem was examined. Both the number of Gaussians per state and useful dimensions per projection in a multiple HLDA system were optimized. The number of useful dimensions to be considered is in the

Complexity Control			WER%		
#Gauss		#Dim	46 hr	76 hr	297 hr
VarMix	12	Fixed 39	38.2	35.8	-
	12	52	38.0	35.3	-
	16	39	38.2	35.9	34.2
	16	52	$37.8^{\dagger}$	$35.6^{\dagger}$	33.7
	20	39	-	-	34.0
	20	52	-	-	$33.6^{\dagger}$
BIC ($\rho = 0.5$)			36.6	34.9	33.4
BIC ($\rho = 1.0$)			36.9	35.2	33.4
BIC ($\rho = 2.0$)			37.2	35.2	33.6
MPE GFunc			36.7	34.6	33.0
(#Gauss)			(18.34)	(14.52)	(17.54)
(#Dim)			(41.78)	(36.67)	(44.77)

TABLE II

OPTIMIZING #GAUSS AND #DIM FOR 65 TRANSFORM HLDA SYSTEMS, † INDICATES THE MOST COMPLEX SYSTEM FOR BIC AND MPE GFUNC.

range from 28 to 52 for each projection. Table II shows the performances of various multiple HLDA systems after complexity control. The baseline systems used VarMix to tune the number of components per state, and fixed the number of dimensions globally as either 39 or 52 across all projections. For all three subsets increasing the number of components per state only gave small improvement. For the 76 hour set increasing the number of components from 12 to 16 actually degraded the performance of the 52 dimensional configuration by 0.3%. Fixing the number of Gaussians per state and increasing the dimensionality from 39 to 52 further reduced the WER for all three training sets by 0.2%-0.5%. As discussed in section II, there are issues for generally using BIC to optimize multiple system attributes with limited amounts of data. BIC systems were generated for the two smaller training sets, the 46 and 76 hour sub sets. The initial statistics for the model selection were derived from same initial models in Table I. The performance of the most complex systems were marked with a "†" in the table. For the 46 hour set, the most complex VarMix system had a WER of 37.8. Using BIC, the best performance was obtained with $\rho = 0.5$, which selected a system with 49.49 useful dimensions per Gaussian and 19.38 components per state on average. Though, as previously mentioned, BIC has limitations for optimizing

multiple attributes, on this task it still gave a gain of 0.9% over the best VarMix system with a fixed number of useful dimensions. A similar trend was also observed on the 76 hour set, where the best BIC system ($\rho = 0.5$) outperformed the best VarMix system by 0.4%.

The marginalized MPE growth functions was then used to determine both the number of components and dimensions. Table II shows these results, along with the size of system generated. WER reduction was obtained over the VarMix baselines for all three training sets. For the 46 hour training set, the GFunc system had 18.34 components per state and 41.78 dimensions per Gaussian. Compared with the 12 component VarMix baselines, it outperformed the VarMix 39 dimensional system by 1.1% absolute, and the more complex 52 dimensional configuration by 0.9%. The MPE GFunc system approximately matched the performance of the best BIC system ($\rho = 0.5$, 19.38 com and 49.49 dim) with much fewer parameters. WER gains were also obtained for the 76 hour set. Compared with the 16 component baselines, the GFunc systems outperformed the 39 and 52 dimensional systems by 1.3% and 1.0% respectively. The gains over the manually tuned BIC systems were by 0.3%-0.6% absolute. On the 297 hour full set, compared with the two 20 Gaussian baselines, there were performance gains of of 0.6%-1.0% absolute. The GFunc system also outperformed all three BIC systems by 0.4%-0.6%. It is interesting that there was no tuning of the MPE GFunc system either between the size of the training data, or the nature of the attributes being optimized.

Set	Complexity Control				WER%	
	#Gauss		#Dim		MLE	MPE
76hr	VarMix	12	Fixed	52	35.3	32.5
	BIC	15.57	BIC	49.36	34.9	32.4
	GFunc	14.52	GFunc	36.67	34.6	31.9
297hr	VarMix	16	Fixed	52	33.7	30.1
	BIC	19.21	BIC	50.91	33.4	29.9
	GFunc	17.54	GFunc	44.77	33.0	29.4

TABLE III

MPE TRAINING OF 65 TRANSFORM HLDA SYSTEMS ON 76 HOUR `h5etrain03sub` AND 297 HOUR `h5etrain03`

All the MPE GFunc systems so far discussed were trained using the ML criterion, although discriminative statistics were used to select the optimal structural configuration. In the next set of experiments, after determining the optimal model structure, model parameters were updated discriminatively using

the standard MPE training [17], [18]. The aim was to investigate the interaction between discriminative training and complexity control. MPE training was only performed on the larger training sets, 76 hours and 297 hours, as these were expected to benefit most from discriminative training. Four iterations of MPE training were performed for each system updating the state transitions, component means, variances and priors only, the multiple HLDA transforms were kept fixed. Based on the WER performances, for the 76 hour set the 12 component 52-dimensional VarMix and the best BIC system ($\rho = 0.5$) in Table II were selected as baselines. Gains from the GFunc systems were mostly maintained after MPE training. However, some gain from the most complex BIC system on the 76 hour set was lost. This may be because compact systems are often preferred for MPE training to ensure good generalization [18]. For this reason, on the 297 hour set the more compact 16 component 52-dimensional VarMix system in Table II was selected as the baseline, instead of the 20 component system with a very similar error rate. The best BIC system ($\rho = 0.5$) was also selected. Again the gain from the GFunc system was mostly additive to MPE training.

VII. Conclusion

A novel automatic model complexity control technique using marginalized discriminative growth functions has been proposed. This discriminative growth function is closely related to a discriminative training criterion, in the work presented here the minimum phone error (MPE) criterion, but has a reduced sensitivity to outliers utterances. To make the marginalization of the function tractable, an EM approach to yield a lower bound approximation is described. This lower bound was then marginalized efficiently using Laplace's approximation for complexity control. This new automatic model complexity scheme was compared with various standard approaches, including the Bayesian Information Criterion (BIC), on a standard LVCSR task. The attributes of the system that were determined were: the number of Gaussians per state, and the number of useful dimensions of an HLDA system. Experimental results showed that this form of discriminative model complexity control gave better, or at least the same performance, of the best manually tuned system. In addition, the approach yields more compact systems than schemes such as BIC. This is useful for speech recognition systems as the gains from discriminative training tend to be larger for more compact systems. Thus this technique is particularly useful for state-of-art speech recognition systems.

References

[1] A. R. Barron, J. J. Rissanen & B. Yu (1998). The Minimum Description Length Principle in Coding and Modeling, *IEEE Transactions on Information Theory*, pp. 2743–2760, Vol. 44, No. 6, October 1998.

[2] W. Chou & W. Reichl (1999). Decision Tree State Tying Based on Penalized Bayesian Information Criterion, *Proc. ICASSP'99*, Vol. 1, Phoenix.

[3] G. Evermann, H. Y. Chan, M. J. F. Gales, T. Hain, X. Liu, D. Mrva, L. Wang & P. C. Woodland, Development of the 2003 CU-HTK Conversational Telephone Speech Transcription System, *Proc. ICASSP'04*, Montreal, Canada.

[4] M. J. F. Gales (2002). Maximum Likelihood Multiple Subspace Projection Schemes for Hidden Markov Models, *IEEE Transactions on Speech and Audio Processing*, pp. 37–47, Vol. 10, 2002.

[5] Z. Ghahramani & M. J. Beal (2000). Graphical Models and Variational Methods, *Advanced Mean Field Method—Theory and Practice*. MIT Press 2000.

[6] V. Goel, S. Kumar, & W. Byrne (2004). Segmental minimum Bayes-risk decoding for automatic speech recognition. *IEEE Transactions on Speech and Audio Processing*, To appear.

[7] P. S. Gopalakrishnan, D. Kanevsky, A. Nadas, and D. Nahamoo (1989). Generalization of the Baum algorithm to Rational Objective Functions, *Proc. ICASSP'89*, pp. 631-634.

[8] J. Kaiser, B. Horvat & Z. Kacic, A Novel Loss Function for the Overall Risk-criterion Based Discriminative Training of HMM Models, *ICSLP'2000*.

[9] X. Liu & M. J. F. Gales (2003). Automatic Model Complexity Control Using Marginalized Discriminative Growth Functions, *Proc. ASRU'03*, St. Thomas, U.S. Virgin Islands.

[10] X. Liu, M. J. F. Gales & P. C. Woodland (2003). Automatic Complexity Control for HLDA Systems, *Proc. ICASSP'03*, Vol. 1, Hong Kong.

[11] X. Liu & M. J. F. Gales (2004). Model Complexity Control And Compression Using Discriminative Growth Functions, *Proc. ICASSP'04*, Montreal.

[12] X. Liu & M. J. F. Gales (2004). *Automatic Model Complexity Control Using Marginalized Discriminative Growth Functions*, Cambridge University Engineering Department Technical Report, CUED/F-INFENG/TR-490, August 2004.

[13] A. Nádas (1983). A Decision Theorectic Formulation of a Training Problem in Speech Recognition and a Comparision of Training by Unconditional Versus Conditional Maximum Likelihood. *IEEE Transactions on Speech and Audio Processing*, pp. 814–817, Vol. 31, 1983.

[14] R. M. Neal (1993) Probabilistic Inference using Markov Chain Monte Carlo Methods, *Technical Report, CGT-TR-93-1*, Department of Computer Science, University of Toronto, 1993.

[15] Y. Normandin (1991). *Hidden Markov Models Maximum Mutual Information Estimation and the Speech Recognition Problem*, PhD thesis, McGill University, Canada.

[16] M. Padmanabhan & L. R. Bahl (2000). Model Complexity Adaptation Using a Discriminant Measure, *IEEE Transactions on Speech and Audio Processing*, pp. 205–208, Vol. 8, No. 2, March 2000.

[17] D. Povey & P. C. Woodland (2002). Minimum Phone Error and I-smoothing for Improved Discriminative Training, *Proc. ICASSP'02*, Florida, USA.

[18] D. Povey (2003). *Discriminative Training for Large Vocabulary Speech Recognition*, PhD thesis, Cambridge University Engineering Department, England.

[19] J. J. Rissanen (1996). Fisher Information and Stochastic Complexity, *IEEE Transactions on Information Theory*, pp. 40–47, Vol. 42, No. 1, January 1996.

[20] G. Schwartz (1978). Estimating the Dimension of a Model, *The Annals of Statistics*, pp. 461–464, Vol. 6, No. 2, February 1978.

[21] K. Shinoda & T. Watanabe (1995). Speaker Adaptation with Autonomous Model Complexity Control by MDL Principle, *Proc. ICASSP'96*, Atlanta, USA.

[22] K. Shinoda & T. Watanabe (1997). Acoustic Modeling based on the MDL Principle for Speech Recognition, *Proc. Eurospeech'97*.

[23] K. Shinoda & T. Watanabe (2000). MDL Based Context Dependent Subword Modeling for Speech Recognition, *Journal of Acoustic Society of Japan*, vol. 21, pp. 79–86, 2000.

[24] V. Valtchev (1995). *Discriminative Methods for HMM-based Speech Recognition*, PhD thesis, Cambridge University Engineering Department, England.

[25] S. Watanabe, Y. Minami, A. Nakamura, N. Ueda (2004) Variational Bayesian Estimation and Clustering for Speech Recognition, *IEEE Transactions on Speech and Audio Processing*, pp: 365–381, Vol. 12, 2004.

[26] A. H. Welsh (1996). Aspects of Statistical Inference, John Wiley & Sons, Inc., 1996.

[27] P. C. Woodland & D. Povey (2002). Large Scale Discriminative Training of Hidden Markov Models for Speech Recognition. *Computer Speech and Language*, Vol. 16, pp. 25-47.

APPENDIX

Following the definition of the MPE criterion in (9), the growth function in (14) may be re-written as

$$\begin{aligned}\mathcal{G}(\lambda) &= \sum_{\tilde{\mathcal{W}}} p(\mathcal{O},\tilde{\mathcal{W}}|\lambda)\mathcal{A}(\tilde{\mathcal{W}},\mathcal{W}) - \mathcal{F}_{\texttt{mpe}}(\tilde{\lambda})p(\mathcal{O}|\lambda) \\ &\quad + C\sum_{\substack{\tilde{\mathcal{W}} \\ \mathcal{A}(\tilde{\mathcal{W}},\mathcal{W})<\mathcal{F}_{\texttt{mpe}}(\tilde{\lambda})}} p(\mathcal{O},\tilde{\mathcal{W}}|\lambda)\left[\mathcal{F}_{\texttt{mpe}}(\tilde{\lambda}) - \mathcal{A}(\tilde{\mathcal{W}},\mathcal{W})\right] \end{aligned} \quad (28)$$

An important aspect of the growth function is its expansion, $\mathcal{G}(\theta,\lambda)$, over hidden variable sequences, $\{\theta\}$. Following (28) above, this is given by

$$\begin{aligned}\mathcal{G}(\theta,\lambda) &= \sum_{\tilde{\mathcal{W}}} p(\mathcal{O},\theta,\tilde{\mathcal{W}}|\lambda)\mathcal{A}(\tilde{\mathcal{W}},\mathcal{W}) - \mathcal{F}_{\texttt{mpe}}(\tilde{\lambda})p(\mathcal{O},\theta|\lambda) \\ &\quad + C\sum_{\substack{\tilde{\mathcal{W}} \\ \mathcal{A}(\tilde{\mathcal{W}},\mathcal{W})<\mathcal{F}_{\texttt{mpe}}(\tilde{\lambda})}} p(\mathcal{O},\theta,\tilde{\mathcal{W}}|\lambda)\left[\mathcal{F}_{\texttt{mpe}}(\tilde{\lambda}) - \mathcal{A}(\tilde{\mathcal{W}},\mathcal{W})\right] \end{aligned} \quad (29)$$

All the following derivation will be based on various forms of the expansion in (29). To make the growth function marginalization more efficient, a lower bound of $\mathcal{G}(\lambda)$ may be derived using an EM like approach via Jensen's' inequality. In a similar fashion a distribution over the hidden state sequences, $\mathcal{P}(\theta,\tilde{\lambda})$, is required. The lower bound is given by

$$\begin{aligned}\log\mathcal{G}(\lambda) &= \log\sum_{\theta}\mathcal{P}(\theta,\tilde{\lambda})\frac{\mathcal{G}(\theta,\lambda)}{\mathcal{P}(\theta,\tilde{\lambda})} \\ &\geq \sum_{\theta}\mathcal{P}(\theta,\tilde{\lambda})\log\frac{\mathcal{G}(\theta,\lambda)}{\mathcal{P}(\theta,\tilde{\lambda})} \\ &= \mathcal{L}_{\texttt{mpe}}(\lambda,\tilde{\lambda}) \end{aligned} \quad (30)$$

In order to make the above bound valid, the hidden variable sequence "posterior" distribution $\mathcal{P}(\theta,\tilde{\lambda})$ must satisfy the non-negative and sum-to-one constraint. The form of posterior considered here is

$$\mathcal{P}(\theta,\tilde{\lambda}) = \frac{\mathcal{G}(\theta,\tilde{\lambda})}{\sum_{\theta}\mathcal{G}(\theta,\tilde{\lambda})} \quad (31)$$

Note that $\mathcal{P}(\theta,\tilde{\lambda})$ is not the true hidden state sequence posterior as used the standard EM algorithm for ML training. Nevertheless it may still be related to a term, $\gamma_{\theta}^{\texttt{mpe}}(\mathcal{O})$, which may be viewed as the MPE hidden state sequence "occupancy". Following (29), this is given by,

$$\mathcal{G}(\theta,\tilde{\lambda}) = p(\mathcal{O}|\tilde{\lambda})\gamma_{\theta}^{\texttt{mpe}}(\mathcal{O}) \quad (32)$$

and

$$\begin{aligned}\gamma_\theta^{\mathtt{mpe}}(\mathcal{O}) &= \sum_{\tilde{\mathcal{W}}} P(\theta,\tilde{\mathcal{W}}|\mathcal{O},\tilde{\lambda})\mathcal{A}(\tilde{\mathcal{W}},\mathcal{W}) - \mathcal{F}_{\mathtt{mpe}}(\tilde{\lambda})P(\theta|\mathcal{O},\tilde{\lambda}) \\ &\quad + C\sum_{\substack{\tilde{\mathcal{W}} \\ \mathcal{A}(\tilde{\mathcal{W}},\mathcal{W})<\mathcal{F}_{\mathtt{mpe}}(\tilde{\lambda})}} P(\theta,\tilde{\mathcal{W}}|\mathcal{O},\tilde{\lambda})\left[\mathcal{F}_{\mathtt{mpe}}(\tilde{\lambda}) - \mathcal{A}(\tilde{\mathcal{W}},\mathcal{W})\right] \end{aligned} \tag{33}$$

When C is big enough the non-negative and sum-to-one constraint will hold for $\mathcal{P}(\theta,\tilde{\lambda})$. To further re-arrange the lower bound in (30), another form of $\mathcal{G}(\theta,\lambda)$, given in (29), is required. This is given by

$$\begin{aligned}\mathcal{G}(\theta,\lambda) &= p(\mathcal{O},\theta|\lambda)\left\{\sum_{\tilde{\mathcal{W}}} P(\tilde{\mathcal{W}}|\theta)\mathcal{A}(\tilde{\mathcal{W}},\mathcal{W}) - \mathcal{F}_{\mathtt{mpe}}(\tilde{\lambda})\right. \\ &\quad \left. + C\sum_{\substack{\tilde{\mathcal{W}} \\ \mathcal{A}(\tilde{\mathcal{W}},\mathcal{W})<\mathcal{F}_{\mathtt{mpe}}(\tilde{\lambda})}} P(\tilde{\mathcal{W}}|\theta)\left[\mathcal{F}_{\mathtt{mpe}}(\tilde{\lambda}) - \mathcal{A}(\tilde{\mathcal{W}},\mathcal{W})\right]\right\} \end{aligned} \tag{34}$$

because for HMMs given the state sequence, the likelihood of observations are independent of the words.

$$p(\mathcal{O},\theta,\tilde{\mathcal{W}}|\lambda) = p(\mathcal{O},\theta|\lambda)P(\tilde{\mathcal{W}}|\theta) \tag{35}$$

Now following (31), (32), and (34), the lower bound in (30) may be re-arranged as

$$\begin{aligned}\mathcal{L}_{\mathtt{mpe}}(\lambda,\tilde{\lambda}) &= \log\mathcal{G}(\tilde{\lambda}) + \sum_\theta \frac{\gamma_\theta^{\mathtt{mpe}}(\mathcal{O})}{\sum_\theta \gamma_\theta^{\mathtt{mpe}}(\mathcal{O})}\log p(\mathcal{O},\theta|\lambda) \\ &\quad - \sum_\theta \frac{\gamma_\theta^{\mathtt{mpe}}(\mathcal{O})}{\sum_\theta \gamma_\theta^{\mathtt{mpe}}(\mathcal{O})}\log p(\mathcal{O},\theta|\tilde{\lambda}) \end{aligned} \tag{36}$$

and the only term associated with model parameters, λ, is given by,

$$\begin{aligned}\sum_\theta \gamma_\theta^{\mathtt{mpe}}(\mathcal{O})\log p(\mathcal{O},\theta|\lambda) &= \sum_\theta \gamma_\theta^{\mathtt{mpe}}(\mathcal{O})\log p(\mathcal{O}|\theta,\lambda) \\ &\quad + \sum_\theta \gamma_\theta^{\mathtt{mpe}}(\mathcal{O})\log P(\theta|\lambda) \end{aligned} \tag{37}$$

For the complexity control problem considered in this paper, the state transition probabilities and Gaussian component priors are kept fixed. Hence the term related to the hidden state sequence priors in (36), $\sum_\theta \gamma_\theta^{\mathtt{mpe}}(\mathcal{O})\log P(\theta|\lambda)$ may be canceled out by $\sum_\theta \gamma_\theta^{\mathtt{mpe}}(\mathcal{O})\log P(\theta|\tilde{\lambda})$. Now the only term related to model parameters, λ, in (36) is $\sum_\theta \gamma_\theta^{\mathtt{mpe}}(\mathcal{O})\log p(\mathcal{O}|\theta,\lambda)$. For HMMs, rather than using the state sequence posteriors, the hidden state occupancies are normally used. The aim is to to re-express the hidden state sequence posteriors, $\gamma_\theta^{\mathtt{mpe}}(\mathcal{O})$, given in (33), as the state occupancies given in (19). To do so $\gamma_\theta^{\mathtt{mpe}}(\mathcal{O})$

needs to be re-written using the MPE word sequence occupancy defi ned in (20). This is given by [6],

$$\gamma_{\theta}^{\texttt{mpe}}(\mathcal{O}) = \sum_{\tilde{\mathcal{W}},\gamma_{\tilde{\mathcal{W}}}^{\texttt{mpe}}\geq 0} P(\theta|\mathcal{O},\tilde{\mathcal{W}},\tilde{\lambda})\gamma_{\tilde{\mathcal{W}}}^{\texttt{mpe}} + \sum_{\tilde{\mathcal{W}},\gamma_{\tilde{\mathcal{W}}}^{\texttt{mpe}}<0} P(\theta|\mathcal{O},\tilde{\mathcal{W}},\tilde{\lambda})\gamma_{\tilde{\mathcal{W}}}^{\texttt{mpe}} - C\sum_{\tilde{\mathcal{W}},\gamma_{\tilde{\mathcal{W}}}^{\texttt{mpe}}<0} P(\theta|\mathcal{O},\tilde{\mathcal{W}},\tilde{\lambda})\gamma_{\tilde{\mathcal{W}}}^{\texttt{mpe}} \quad (38)$$

now for HMMs by summing over all the sequences passing through the same state for each time instance, the MPE statistics, $\gamma_j^{\texttt{mpe}}(\tau)$, in (19) may be derived, and one may also write

$$\sum_{\theta}\gamma_{\theta}^{\texttt{mpe}}(\mathcal{O})\log p(\mathcal{O}|\theta,\lambda) = \sum_{j,\tau}\gamma_j^{\texttt{mpe}}(\tau)\log p(\boldsymbol{o}_\tau|\theta_\tau = s_j,\lambda)$$

which is the MPE auxiliary function, $\mathcal{Q}_{\texttt{mpe}}(\lambda,\tilde{\lambda})$, in (17). Finally, given this form of $\mathcal{Q}_{\texttt{mpe}}(\lambda,\tilde{\lambda})$ the growth function lower bound in (36) may be re-written as in (16).

[6]Note the binary partition of all possible word sequences with respect to the sign of $\gamma_{\tilde{\mathcal{W}}}^{\texttt{mpe}}$ was also used in the standard form of MPE statistics of (22) as proposed in [18] for discriminative training.