Cluster Adaptive Training With Factorized Decision Trees For Speech Recognition

Kai Yu and Hainan Xu

Institute of Intelligent Human-Machine Interaction
MOE-Microsoft Key Lab. of Intelligent Computing and Intelligent Systems
Department of Computer Science and Engineering
Shanghai Jiao Tong University, 200240, Shanghai, P. R. China
Email:{kai.yu,xhnwww}@sjtu.edu.cn

Abstract

Cluster adaptive training (CAT) is a popular approach to train multiple-cluster HMMs for fast speaker adaptation in speech recognition. Traditionally, a cluster-independent decision tree is shared among all clusters, which could limit the modelling power of multiple-cluster HMMs. In this paper, each cluster is allowed to have its own decision tree. The intersections between the triphones subsets, corresponding to the leaf nodes of each cluster-dependent trees, are used to define a finer state sharing structure. The parameters of these intersections are constructed from the parameters of the leaf nodes of each individual decision tree. This is referred to as *CAT with factorized decision trees* (FD-CAT). FD-CAT significantly increases the modelling power without introducing additional free parameters. A novel iterative mean cluster update approach and a robust covariance matrix update method with united statistics are proposed to efficiently train FD-CAT. Experiments showed that using multiple decision trees can yield better performance than single decision tree. Furthermore, FD-CAT significantly outperformed traditional CAT system.

Index Terms Adaptation, factorized decision trees, cluster adaptive training, eigenvoices

1. Introduction

Cluster based adaptation has been widely used for fast speaker adaptation in speech recognition. It performs adaptation on a series of sets of HMMs. One traditional scheme is to build speaker dependent models and choose an appropriate one for a particular speaker. In this approach, the adaptation, or the selection of the appropriate model for recognition is a "hard" choice. Alternatively, linear combination can be used to construct a new interpolated model from a set of speaker dependent models [1], which is a "soft" choice. Cluster adaptive training [2] and eigenvoices [3] are both based on the soft choice concept. These approaches can be interpreted as adaptive training, where the "transforms" to adapt the model are the interpolation weights. Rather than using a single set of HMMs, cluster adaptation techniques require multiple sets of HMMs.

Cluster adaptive training (CAT) [2, 4] is a general framework to train multiple-cluster HMMs and interpolation weights. The basic idea is to build a target speaker specific model by using a weighted sum of multiple sets of HMMs. In order to simplify training, it is often assumed that different clusters share the same covariance matrices, transition matrices and mixture weights and that only the mean values differ between the clusters. A similar method is *eigenvoices* [5], where multiple-cluster HMMs are constructed by performing eigen-decomposition on a set of speaker-dependent concatenated HMM parameters. Previous works on CAT always use a single decision tree for the multiple-HMMs. In this case, eigenvoices is equivalent to CAT with eigen-decomposition initialization. However, the difference between CAT and eigenvoices does not only lie in the initialization stage. With explicit association with different acoustic factors, it is possible to employ separate parameter tying structure for each cluster within the CAT framework. This will allow more powerful modelling without dramatically increasing the number of free parameters. This is a distinct advantage of the CAT framework compared to eigenvoices.

The use of multiple dependent decision trees in multi-cluster HMMs has been explored in two different frameworks [6, 7]. In [6], acoustic scores given by different clusters are linearly combined to give the final score. Another framework is to combine the effect at parameter level. This framework has been used within the adaptive training framework in HMM based speech synthesis to model rich contexts [7]. Instead of combining acoustic scores as proposed in [6], the Gaussian parameters of the atomic cluster are constructed among parameters associated with the leaf nodes of the multiple decision trees.

This paper investigates *CAT with factorized decision trees* (FD-CAT) for speech recognition. Due to the combination effect from intersection, the update formula for multiple-cluster HMMs in traditional CAT are no longer applicable. An extended formulae has been proposed in [8], where multiple cluster means are updated simultaneously. To avoid computational cost and numerical accuracy issue of simultaneous update, a new iterative update of multiple-cluster mean vectors is proposed as an alternative. An important issue of FD-CAT is how to reliably update the covariance matrices associated with the atomic state clusters. A united statistics approach is proposed to effectively address the issue.

The rest of the paper is arranged as follows. Section 2 describes the framework of FD-CAT and the new parameters update approaches. Experiments are presented in section 3, followed by conclusions and relations to prior works.

2. CAT with factorized decision trees

In cluster adaptive training (CAT) [2], the canonical model is a set of multiple-cluster HMMs. Here, *multiple clusters* are represented using a set of P mean vectors of each Gaussian com-

This research was partly supported by the Program for Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning and the China NSFC project No. 61222208.

ponent m, one for each of the P clusters. In traditional CAT, the mixture weight, $c^{(m)}$ and the covariance matrix (usually diagonal). $\mathbf{\Sigma}^{(m)}$ are shared among all clusters, as the multiple-cluster HMMs are initialized from a set of standard cluster-independent HMMs [9] (or a set of cluster-dependent HMMs in the case of eigenvoices [3]). Therefore, it naturally inherits the state sharing structure, i.e. all clusters sharing use the same decision tree as the seed HMM model. The adapted mean of Gaussian component m for a particular speaker s, $\hat{\boldsymbol{\mu}}^{(sm)}$, can then be expressed as

$$\hat{\boldsymbol{\mu}}^{(sm)} = \sum_{i=1}^{P} \lambda_i^{(sm)} \boldsymbol{\mu}_i^{(m)} \quad (1)$$

where $\boldsymbol{\mu}_i^{(m)}$ is the mean of the i^{th} cluster in Gaussian component m, P is the total cluster number. Note that, since all clusters use the same decision tree and Gaussian ordering structure, the superscript for the adapted mean, m, is the same for all clusters. $\lambda_i^{(sm)}$ is the corresponding weight for Gaussian component m and speaker s. It is possible to construct a regression tree [10] for interpolation weights to allow flexible adaptation, so that a group of Gaussians may share the same interpolation weight vector. The regression base-class notation will be omitted in the derivation in this section for clarity.

2.1. CAT with factorized decision trees

Though CAT has achieved good performance in fast speaker adaptation, the use of single decision tree is a limitation. In the *factorized decision tree* approach, separate decision trees are constructed individually for different context groups. The intersections of leaf nodes of different trees are then used as the atomic state sharing units. However, the actual Gaussian parameters associated with each atomic leaf node are not tied at the intersection level, but calculated from leaf nodes of each clusters. Adaptive training framework can then be employed to interleavingly update the multiple sets of parameters [7]. CAT has been used within this framework for a single Gaussian system in HMM based speech synthesis [8]. It has been shown that it can effectively improve the modelling of rich contexts [8, 7].

In this paper, the framework is extended to acoustic model adaptation in speech recognition, referred to as *CAT with Factorized Decision trees* (FD-CAT). Here, acoustic factors like gender or environment noise type, are considered as effective contexts during state clustering. Separate decision trees are built for each acoustic factor. The intersections of these decision trees are then calculated and defined as the atomic leaf nodes. This operation is referred to as *combination of multiple decision trees*. An example of combining two decision trees is depicted in Fig 1.

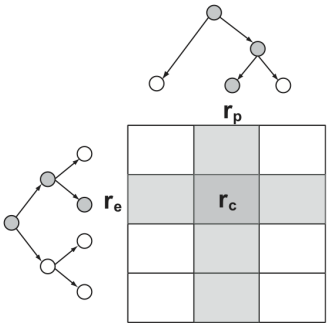

Figure 1: Combination of two decision trees

Assuming the triphones clustered into leaf nodes r_p and r_e are

$$r_p = \{\texttt{a-b+c, d-b+c, e-b+f, c-b+f}\}$$
$$r_e = \{\texttt{e-b+c, d-b+c, e-b+a, c-b+f}\}$$

Then the intersection of the two leaf nodes, r_c, in the combined decision tree is

$$r_c = r_p \bigcap r_e = \{\texttt{d-b+c, c-b+f}\}$$

Due to the combination effect, the number of the atomic leaf nodes r_c is normally significantly greater than the total number of the leaf nodes in the two decision trees. In general, the atomic leaf node r_c of the combined decision tree is the intersection of the leaf nodes corresponding to each individual decision tree. The general case can be expressed as

$$r_c = \bigcap_{i=1,\cdots,P} r_d^i \quad (2)$$

where i is index of cluster (acoustic factor), r_d^i denotes the leaf node of the i^{th} decision tree. With the factorized decision trees, the mean of Gaussian component m in atomic state r_c can be expressed as[1]

$$\hat{\boldsymbol{\mu}}^{(s\theta_c)} = \sum_{i=1}^{P} \lambda_i^{(s\theta_c)} \boldsymbol{\mu}^{\theta_d^i} \quad (3)$$

where $\theta_c = \{r_c, m\}$ is a brief notation for the m^{th} Gaussian of the atomic leaf node r_c in equation (2), similarly, $\theta_d^i = \{r_d^i, m\}$ denotes the m^{th} Gaussian of the leaf node of the clustered state of the i^{th} decision tree r_d^i. Here, $r_c \subseteq r_d^i$. P is the total number of clusters (acoustic factors). For example, *gender* may be used as the acoustic factor, so that there are 2 clusters corresponding to *female* and *male* respectively. s denotes a distinct acoustic condition such as a speaker. Each speaker is associated with different weights for male and female clusters. $\hat{\boldsymbol{\mu}}^{(s\theta_c)}$ is the adapted mean of the m^{th} Gaussian component within the atomic state r_c. It is constructed from the means of the m^{th} Gaussian component within each individual decision tree, $\boldsymbol{\mu}_i^{\theta_d^i}$. The construction is done by linear interpolation using a set of weights. $\lambda_i^{(s\theta_c)}$ is the weight for gender cluster i, which is associated with speaker s and Gaussian component θ_c, i.e., each Gaussian component in the atomic leaf node may have a different set of weights. This allows more flexible and powerful modelling. To avoid data sparsity issue, it is a common practice to group the Gaussian components using a regression classes tree [10] and associate the weights to each node of the regression tree according to the amount of available data.

Due to the combination effect, the number of parameters of the adapted mean can be significantly larger than the summation of the free parameters on the right-hand-side of the equation. Hence, by employing factorized decision trees, finer state modelling can be easily achieved. On the other hand, with the structured parametric representation, robust parameter estimation is also possible given appropriate decision tree clustering thresholds.

An important difference of equation (3) from equation (1) is that the mean vector of each decision tree can contribute to

[1]Note that, for state output distribution with Gaussian mixture models (GMM), not only the decision tree, but also the Gaussian ordering can have overlapping structures. This means that the construction of the parameters of the m^{th} Gaussian component of r_c can involve the n^{th} Gaussian component of r_d^i. In this paper, only the state sharing structures are considered and the Gaussian ordering is assumed to be unchanged.

several different atomic clusters. Due to this overlapping effect, it is not possible to directly update the CAT model parameters of a single atomic state as in [9]. Instead, all overlapped mean vectors have to be updated dependently. Although there is a closed form solution for the simultaneous update[8], it requires inversion of very large matrices and may not be stable during training. To efficiently apply FD-CAT to speech recognition, an alternative approach will be discussed in the next section.

As for the interpolation weights update, there is no difference between FD-CAT and traditional CAT. The weights update formula in [9] can be used without any change.

2.2. Iterative mean update of FD-CAT

To avoid computational cost and the numerical accuracy issue of large matrix inversion, mean vectors of each cluster may be updated sequentially rather than simultaneously. The auxiliary function of FD-CAT for mean update can be written as

$$\mathcal{Q} = -\frac{1}{2}\sum_{s,t,\theta_c}\gamma_{\theta_c}(t)\big(\mathbf{o}_t-\hat{\boldsymbol{\mu}}^{(s\theta_c)}\big)^T\boldsymbol{\Sigma}^{(\theta_c)-1}\big(\mathbf{o}_t-\hat{\boldsymbol{\mu}}^{(s\theta_c)}\big) \quad (4)$$

where θ_c and $\hat{\boldsymbol{\mu}}^{(s\theta_c)}$ are defined in equation (3), $\gamma_{\theta_c}(t)$ is the posterior occupancy of θ_c at time t. Let $\boldsymbol{\mu}^{\theta^i_d}$ be the mean vector of the m^{th} component of the leaf node r_d of the i^{th} decision tree, differentiating (4) w.r.t. $\boldsymbol{\mu}^{\theta^i_d}$ and equating it to zero leads to the below mean update formula:

$$\boldsymbol{\mu}^{\theta^i_d} \;=\; \mathbf{G}^{-1}_{\theta^i_d}\mathbf{k}_{\theta^i_d} \quad (5)$$

where $\mathbf{G}^{-1}_{\theta^i_d}$ and $\mathbf{k}_{\theta^i_d}$ are sufficient statistics defined as

$$\mathbf{G}_{\theta^i_d} = \sum_{s,t,r_c\subset r_d}\gamma_{\theta_c}(t)\big(\lambda^{(s\theta_c)}_i\big)^2\boldsymbol{\Sigma}^{(\theta_c)-1}$$

$$\mathbf{k}_{\theta^i_d} = \sum_{s,t,r_c\subset r_d}\gamma_{\theta_c}(t)\lambda^{(s\theta_c)}_i\boldsymbol{\Sigma}^{(\theta_c)-1}\Big(\mathbf{o}_t-\sum_{j\neq i}\lambda^{(s\theta_c)}_j\boldsymbol{\mu}^{\theta^j_d}_j\Big)$$

From equation (5), the mean vector associated with r^i_d is dependent on the parameters of the other clusters $j = 1,\cdots,P,\ j\neq i$. By interleaving this update process across all clusters, the mean clusters of FD-CAT can be effectively updated sequentially.

2.3. Covariance matrix update of FD-CAT

By taking intersections between different decision trees, FD-CAT can significantly increase the number of atomic clustered states and consequently, Gaussian components. This will not affect the robustness for mean vectors update because mean vectors are associated with the original decision trees where sufficient data is guaranteed during state clustering. However, covariance matrices in FD-CAT are associated with atomic Gaussian components. The re-estimation only relies on the data aligned to the atomic states during expectation maximization (EM) training. This is referred to as *intersected statistics*. The large number of atomic Gaussian components may lead to sparse data and unreliable covariance estimation.

One way to address this issue is to build a new regression tree for covariance matrices, which is mentioned in [8]. By sharing covariance matrix parameters of each node, robust estimation can be obtained. However, it is not easy to find an ideal regression tree as the build of regression tree requires a well trained seed model. In this paper, an alternative approximation approach, *united statistics for covariance update*, is used. The basic idea is to use the statistics of all clusters which overlap with the atomic intersection leaf node, to update covariance matrix. With the notation in (2), the covariance update with united statistics can be written as

$$\boldsymbol{\Sigma}^{(\theta_c)} \;=\; \mathtt{diag}\Bigg(\frac{\sum_{i=1}^{P}\sum_{r_c\subset r^i_d}\sum_{s,t}\gamma_{\theta_c}(t)\tilde{\mathbf{o}}^T_t\tilde{\mathbf{o}}_t}{\sum_{i=1}^{P}\sum_{r_c\subset r^i_d}\sum_{s,t}\gamma_{\theta_c}(t)}\Bigg) \quad (6)$$

where $\mathtt{diag}(\cdot)$ takes diagonal elements, $\theta_c = r_c(m)$ is brief notation Gaussian component m of atomic state r_c, and

$$\tilde{\mathbf{o}}_t \;=\; \mathbf{o}_t-\hat{\boldsymbol{\mu}}^{(s\theta_c)} = \mathbf{o}_t-\sum_{i=1}^{P}\lambda^{(s\theta_c)}_i\boldsymbol{\mu}^{\theta^i_d}_i \quad (7)$$

From equation (6), the statistics of each individual cluster are repeatedly used, which will effectively address the data sparsity issue. Note that, this update is only an approximation rather than an exact solution derived using the EM algorithm.

3. Experimental results

The performance of FD-CAT was evaluated on a Wall Street Journal (WSJ0) large vocabulary speech recognition task [11]. The SI-84 training set was used as the training corpus. It consists of 83 speakers (41 male and 42 female) and 7138 utterances, giving a total of about 14 hours speech. The standard `nov92` 5K vocabulary non-verbalized test set was used for evaluation. It consists of 8 speakers (5 male and 3 female), each with about 40 utterances. All systems had 16 Gaussian components per state on average, and used MFCC front-end with energy and their first, second derivatives, resulting in 39 dimensional feature vector. All systems were cross-word triphone systems. A bi-gram language model was used in decoding.

The baseline systems were built using a single decision tree with 3118 clustered states, constructed on all training data, referred to as a *GI decision tree*. A gender-independent (GI) and a gender-dependent (GD) system were built with the maximum likelihood (ML) criterion. During testing, gender information was assumed to be known for the GD system. Hence, the GD performance shown in this section is slightly optimistic. With optimistic estimation of the GD performance, the conclusion of this paper is even sounder since in CAT the gender labels of test data are always assumed to be unknown. To achieve more robust estimate, maximum a posteriori (MAP) criterion was also used to build a MAP-GD system for comparison.

The above baseline systems all used the GI decision tree. It is also interesting to investigate the effect of using multiple decision trees. Two new decision trees were then constructed using the female and male training data respectively. As the amount of data was reduced compared to the GI decision tree, the stopping thresholds were tuned so that the resultant numbers of the clustered states are similar to the GI decision tree. In total, there are 3122 clustered states resulting from the female decision tree, and 3111 from the male one. Some initial experiments showed that using the male or female decision tree to replace the GI decision tree can not yield consistent performance gain due to the biased state clustering structure. To get an estimate of the effect of combining the two gender-dependent decision trees, a cheating experiment was performed. The GI/ML-GD/MAP-GD systems were re-trained using the female and the male decision tree respectively, yielding two sets of GI/ML-GD/MAP-GD systems. During the re-training, all procedures were exactly the same as before except for using different state tying structures. Then, the systems trained on the female decision tree were used to decode the female test data, and similarly to

the male counterpart. The hope is that the biased decision tree structure will benefit the decoding on the corresponding type of test data.

Decision Trees		Systems		
		GI	ML-GD	MAP-GD
Single	GI	7.10	7.15	6.95
Multiple	Male + Female	6.93	6.63	6.61

Table 1: WER (%) of baseline GI/GD systems with single or multiple decision trees

Table 1 shows the performance of all baseline systems. It can be observed that, with the single GI decision tree, the ML-GD system got slight degradation due to the reduced amount of training data after gender splitting. Employing the MAP criterion can achieve more robust estimate, and consequently better result than the ML-GI system[2]. When the gender-dependent decision trees were combined and used, the performance of all systems were improved. Although the performance was slightly optimistic due to the assumption of known gender during testing, it still demonstrates that appropriate combination of multiple decision trees can be beneficial. Since the MAP-GD systems yield the best results, they are used as the baseline system to be compared to cluster adaptive training (CAT) systems.

A traditional CAT system was built using the GI decision tree. It was initialized from the GI system using gender information from training data [9]. When multiple decision trees (female and male) were used, a FD-CAT system was constructed. As indicated in section 2.1, FD-CAT can yield significantly increased model complexity without increasing the number of free parameters. This can be demonstrated using the number of atomic states and the number of free states. As shown in table 2, compared to the original decision trees, the number of atomic states of factorized decision trees (Male $\cap$ Female) has been significantly increased while the number of free states keeps unchanged.

Decision Trees		#States	
		Atomic	Free
Single	GI	3118	3118
Multiple	Male + Female	6233	6233
	Male $\cap$ Female	8275	6233

Table 2: Comparison of model complexity and free parameters

As indicated in section 2.3, covariance update can be unreliable[3] in FD-CAT. To investigate this issue, normal update of covariance matrix and the union update approach were compared.

In the `normal` covariance update approach, only the statistics corresponding to the atomic state (intersection of leaf nodes) were used for update. `union` denotes the proposed united statistics approach. It can be observed from table 3 that the normal covariance update got significant degradation due to data sparsity. On the other hand, the union covariance update approach achieved more robust estimate, and consequently led to much better performance. Therefore, in the following FD-CAT experiments, the united statistics approach was always used.

Variance Stat.	WER (%)
`normal`	8.22
`union`	6.74

Table 3: Different variance update approach for FD-CAT

As mentioned in section 2, regression tree can be used for interpolation weights in CAT system. A set of traditional CAT and FD-CAT systems with different number of regression base classes for weight vectors were built for comparison.

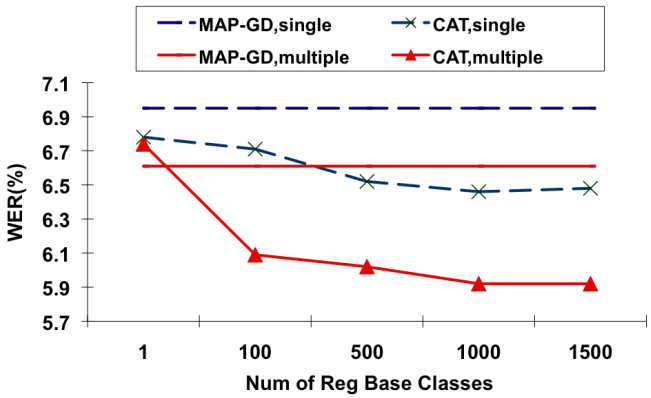

Figure 2: WER (%) comparison of MAP-GD and CAT systems with single and multiple decision trees

Fig 2 shows the comparison between FD-CAT, traditional CAT and the MAP-GD systems. Note that single and multiple decision trees details are shown in table 2. It can be observed that, with either single or multiple decision trees, CAT systems can outperform MAP-GD systems. Systems with multiple decision trees outperformed systems with single decision tree. With more regression base classes, performance of all CAT systems improved. The most important observation is that FD-CAT with more than 100 regression base classes can outperform all traditional CAT systems and MAP-GD systems. Statistical significance test showed that all gains were significant. This effectively demonstrates that FD-CAT is more powerful than traditional CAT as well as MAP-GD.

4. Conclusion

This paper applies the factorized decision trees approach to cluster adaptive training (FD-CAT) for speech recognition. By finding intersections of multiple decision trees, FD-CAT significantly increases the number of atomic states without affecting the number of free states from the original decision trees. This leads to more powerful modelling ability. With an appropriate choice of regression tree, FD-CAT can significantly outperform the traditional CAT as well as the MAP-GD systems.

[2]It is worth noting that the improvement did not come from the increased number of parameters in GD training. A GI system with 32 Gaussian components was also built and the WER was 7.42, showing that the original 16 component system has got converged result.

[3]As component priors are also dependent on the intersected states, their update may also be unreliable. Similar united statistics approach was also used in this paper.

5. References

[1] T. Hazen and J. Glass, "A comparison of novel techniques for instantaneous speaker adaptation," in *Proc. EuroSpeech*, 1997, pp. 2047–2050.

[2] M. J. F. Gales, "Cluster adaptive training for speech recognition," in *Proc. ICSLP*, 1998, pp. 1783–1786.

[3] R. Kuhn, P. Nguyen, J. C. Junqua, L. Goldwasser, N. Niedzielski, S. Fincke, K. Field, and M. Contolini, "Eigenvoices for speaker adaptation," in *Proc. ICSLP*, 1998, pp. 1771–1774.

[4] K. Yu and M. J. F. Gales, "Discriminative cluster adaptive training," *IEEE Transactions on Speech and Audio Processing*, vol. 14, no. 5, pp. 1694–1703, 2006.

[5] 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.

[6] J. Xue and Y. Zhao, "Random forest of phonetic decision trees for acoustic modeling in conversational speech recognition," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 16, pp. 519–828, 2008.

[7] K. Yu, H. Zen, F. Mairesse, and S. Young, "Context adaptive training with factorized decision trees for hmm-based statistical parametric speech synthesis," *Speech Communication*, vol. 53, no. 6, pp. 914–923, 2011.

[8] H. Zen and N. Braunschweiler, "Context-dependent additive $\log F_0$ model for HMM-based speech synthesis," in *Proc. of Interspeech*, 2009, pp. 2091–2094.

[9] M. J. F. Gales, "Cluster adaptive training of hidden Markov models," *IEEE Transactions on Speech and Audio Processing*, vol. 8, pp. 417–428, 2000.

[10] M. J. F. Gales, "The generation and use of regression class trees for MLLR adaptation," Tech. Rep. CUED/F-INFENG/TR263, Cambridge University Engineering Department, 1996.

[11] D. B. Paul and J. M. Baker, "The design of the Wall Street Journal-based CSR corpus," *Proc. DARPA Speech and Natural Language Workshop*, 1992.