Pruning of State-Tying Tree using Bayesian Information Criterion with Multiple Mixtures

Yu-Chung Chan, Manhung Siu, Kan-Wing Mak *

ABSTRACT

The use of context-dependent phonetic units together with Gaussian mixture model allow modern-day speech recognizer to build very complex and accurate acoustic models. However, because of data sparseness issue, some sharing of data across different triphone states is necessary. The acoustic model design is typically done in two stages, namely, designs of the state-tying map and grows the number of mixtures in each tied-state. In the design of the state-tying map, single Gaussian is used to represent the data. This ignores the fact that a single Gaussian is an insufficient model. In this paper, we propose a simple modification to the two stage process by adding a third stage. In this added stage, the state-tying tree is pruned and the pruning is based on the mixture representation of the tied-states. We propose using Bayesian Information Criterion(BIC) as the criterion for this pruning and show that by adding this step, the resulting model is more compact and gives better recognition accuracy on the Resource Management(RM) task.

1. Introduction

Nowadays, most large vocabulary continuous speech recognizers (LVCSR) use triphone-based continuous Hidden Markov Models (HMM) to represent the speech acoustics. The use of context-dependent models (triphones) combines with Gaussian mixture can sufficiently models the complex observation distributions. However, because of the large number of possible triphones, data sparseness becomes an issue. Thus, sharing of data across different, but acoustically similar triphone states is often necesary. This process is often referred as state-tying and the resulting shared states are called clustered-states. How states are tied can have significant impact on the goodness of the model. Different measures of similarity and clustering approaches have been proposed. One widely used top-down state-tying approach is the decision-tree of which the leave nodes define the clustered-states. As the tree grows from the root node (which is modeled by a single Gaussian), the number of clustered-states increases and the model is refined.

While Gaussian mixtures are used to represent the final model, single Gaussian is used to represent the clustered-state during the state-tying process. One reason is that for some similarity measures, such as cross-entropy, it is difficult to obtain a close form solution for comparing Gaussian mixtures. Another reason is that for a top-down clustering algorithm, it is computationally expensive to estimate these mixtures at all steps. The acoustic model design is commonly performed in two stages.

- **State-tying** Similar triphone states, each represented by a single Gaussian, are tied.
- **Mixture growing** For each tied-state, the number of Gaussian components are increased.

In both processes, thresholds on optimality criteria are used to stop the process.

One can view both the top-down tree growing and the mixture growing as refining the acoustic representation of the data. However, this two-step approach can be problematic. First, while both are refining the acoustic representation by increasing the model complexity, their interaction is unidirectional (the state-tying affecting the mixture growing). In fact, the process is done sequentially in which state-tying is finished first making hard decisions about what state to tied before mixtures are grown. Second, the optimality measures used in the two processes are sometimes different which creates an inconsistency.

In this paper, we propose a new approach in performing the state-tying and mixture growing that aims to remedy part of the issue. In this new approach, the state-tying tree can be modified after the mixtures are grown. Instead of a two-step process, we propose a three-step process. After the state-tying tree is built and mixtures are grown for all clustered-states, we add the extra step of pruning the state-tying tree. The pruning is based on the actual mixture representation something better. of the tied states. This is different to [10] which proposed an extra agglomerative clustering be done on the state-tying tree in two aspects. 1) Mixture representation is not taken into account in [10]. 2) A tree-based structure is maintained under our proposed approach while that is not necessarily true in [10].

*Dr. Man-hung Siu and Yu-Chung Chan are with the Department of Electrical and Electronic Engineering, and Dr. Brian Mak is now with the Department of Computer Science the Hong Kong University of Science and Technology, Hong Kong SAR, China.

This paper will be organized as follows. Section 2 gives an overview of the Bayesian information criterion and Sections 3 and 4 describe the decision-tree based state tying and mixture growing respectively. Section 5 describes the proposed method in detail. The experimental results are reported in Section 6 and a summary is given in Section 7.

2. Bayesian Information Criterion (BIC)

Bayesian Information Criterion is defined as $BIC(n) = log(P(O,S|M)) - \frac{\lambda}{2} N_C log(N)$

where $log(P(O,S|M))$ is the log-likelihood of the model, $\mathcal{N}_C$ is the number of parameters used in the mixtures and $\mathcal{N}$ is the number of data points in the mixtures. The parameter λ is the complexity factor which represents the penalty of increasing the complexity of a model.

When using BIC as an optimality criterion for either state-tying or mixture growing, one can view BIC as setting dynamic thresholds that are dependent on the amount of data represented by the model.

It should be noted that determining the optimal model size is a difficult problem and hypothesis testing techniques are used. The derivation of the BIC equation above is based on asymptotic assumptions which may require large amount of data to hold [9].

3. Decision-Tree based State-Tying

As we have discussed before, state-tying is necessary because of the data sparseness problem. Given the large number of triphone states, there can be almost infinite ways of tying the states or partitioning the them into groups. As a result, different clustering algorithms were suggested. While purely data-driven approaches, such as agglomerative clustering of the observed states, can effectively tie observed states, how to handle unseen triphone becomes an issue. Decision-tree based state-tying [6] provides a good compromise between purely data-driven approaches and the use of human knowledge.

The central idea of state-tying using decision-tree is to reduce the possible number of partitions of the data by constraining the state space using prior knowledge such as phonetic knowledge [7]. This knowledge is applied to define a set of possible questions $Q_i(x), Q_2(x), ...Q_n(x)$ about the phonetic context. For implementation simplicity, binary questions are used. Initially, all the data are pooled together at the root node which is also the only leave node. Tentative partitions are formed by asking all possible questions on existing leave nodes. The question and leave node pair which optimizes a global cost function C will be chosen and the data assigned to the selected leave node will be partitioned. This process is repeated until some stopping criterion is meet. For observed triphone states, those share the same leave node are tied together forming a clustered-state. The decision-tree also serves as a mapping between unseen triphone states and clustered-states because the partitioning is based on phonetic questions which are applicable to unobserved triphones. Several design decisions have to be made in order to use decision tree for state-tying. First question is the candidates for tying. To enhance the performance and reduce computation time, triphone states are only shared between those having the same center phone. Thus, a different decision-tree is built for each center phone. Obtaining good linguistic questions for the decision tree is also important and those are usually created manually and change from language to language. Another important issue is defining a suitable cost function. Different criteria are suggested, including the likelihood increment, entropy and BIC. Lastly, the tree growing is stopped when either the tree is grown to its maximum size or no good partition can be found. It is also possible to intentionally overgrown a tree and then prune the tree back. Because decision-tree growing is a greedy algorithm, intentionally over-growing the tree can served as a look-ahead that preserved more branches.

Since the tree growing procedure involves a lot of tentative partitions, the data at each tree node is typically represented by only one single Gaussian. In the next Section, we will discuss ways to increase the number of mixtures in the leave nodes.

4. Mixture Growing

After state-tying, Gaussian mixtures are grown to represent the clustered-states. Typically, mixture growing includes three steps. 1) Create an initial model that have one extra mixture components. One widely used method is to split the Gaussian component with the biggest mixture weight. By perturbing the Gaussian with the addition and subtraction of a fraction of its standard deviation, a Gaussian is split into two. 2) Re-estimate the mixture means and covariances either by the K-means algorithm or the forward-backward re-estimation. 3) Decide whether the new model should be keep by measuring the model using an optimality criterion and compare it with a threshold. Since different clustered-states represent different number of data-points, each clustered-state should be modeled by a different number of mixtures.

In a sense, mixture growing is very similar to the top-down process in state tying. It requires an criterion to determine how to partition the data into separate groups which are represented by different models. In this section, we describe three optimality criteria, 1) the occupancy count, 2) likelihood increment and 3) BIC.

1, Occupancy count: This criterion stops the mixture growing when the number of data assigned to a mixture or the average number of data points per mixture falls below a pre-set minimal. This can also be extended to occupancy count as described in [3, 11]

2, Likelihood Increment : Under the likelihood criterion, the likelihood $P(O|M)$ is computed after the k-means or forward-backward re-estimation. The growing process stops when

$$P(O|M_{n+1}) - P(O|M_n) < T \quad (1)$$

where T is an arbitrary threshold.

3, BIC approach: Similar to likelihood increment, BIC is calculated after the k-mean or forward-backward re-estimation. The growing process stops when

$$BIC(M_{n+1}) - BIC(M_n) > 0, \quad (2)$$

where $BIC(M_k)$ is the BIC for the model with k mixtures. It has been reported in [6] that BIC improves the recognition accuracy when compared to likelihood increment. A discussion of modeling mixture Gaussians using BIC can be found in [7].

5. Proposed Method

As we have briefly mentioned in Section 1, our proposed algorithm aims at integrating the mixture growing process with the state-tying tree design. While it may be best to use multiple mixtures in the state-tying process, that can be computationally expensive. Instead, we propose pruning the state-tying tree based on a criterion measured using multiple mixtures.

The proposed algorithm includes the following steps:

1. Tied triphone states using decision tree as suggested in [10].
2. For all clustered-states, grow the number of mixtures till optimal as described in Section 4,
3. For all parent nodes whose child nodes are all leave nodes (called eligible nodes), grow the number of mixtures till optimal as described in Section 4. Then, compare the BIC of the parent node against the joint BIC of the child nodes. If the parent node has better BIC, i.e.,

$$\begin{aligned} &BIC(parent_{node})_{N_{parent}} \\ &> BIC(left_{node})_{N_l} + BIC(right_{node})_{N_R} \end{aligned} \quad (3)$$

then prune the child nodes. N_{parent}, N_l, N_r denotes the optimal number of mixtures in the parent node, left child node ,right child node respectively. The merging process can be recursively applied until all eligible merges are considered.

BIC is controlled by the parameter λ. By adjusting λ, we can control the degree of merging. The larger λ, the greater the degree of tying. It is worth noting that BIC has been shown to outperform likelihood in both state-tying and mixture growing. If BIC is used for both processes, we can view the combined process as a unified process of finding the best complexity in modeling the data.

6. Experimental Results

Our experiments are performed on the Resource Management (RM) task using the HTK toolkit. The features set includes the mel cepstral coefficients, the log energy, and their first and second order derivatives. Cross-word triphone models are trained using the decision-tree based tying as suggest in [10]. However, we have disabled the extra agglomerative clustering step which is suggested to be applied after the decision tree is built. In all our experiments, the decision tree is built using the log likelihood criterion which produce a tree with 1800 clustered-states.

We performed two sets of experiments. In the first set, we compared different mixture growing criteria, including the occupancy count, likelihood increment and BIC. In the second set of experiments, we compare the effect of merging the tree nodes based on multiple mixture. Since the resulting model contains fewer number of Gaussians, a control experiment is performed such that the number of Gaussians is the same.

6.1. Experiment I: Comparison of Different Mixture Growing criteria

Three criteria have been tested in this experiment using the Feb89 test set.

1, Occupancy approach We assume each Gaussian mixture occupy a 180 of data points. Based on this assumption, the number of Gaussians are incrementally increased until there is insufficient data. Forward-backward re-estimation is performed after each increment.

2, Likelihood Increment approach The mixture is grown until the likelihood stop to exceed a certain threshold. The mixture growing stops if the percentage of likelihood improvement is smaller than 0.5%.

3, BIC approach The mixture is grown until the BIC of the model stops to increase. Only $\gamma = 1$ has been tested here.

Method	Accuracy	Size
Occupancy count	95.59	6249
Likelihood Increment	96.06	7925
BIC	96.25	9901

Table 1: Performance of using different mixture growing criteria on the Feb89 test set

The recognition accuracy using different mixture growing criteria is tabulated in Table 1 along with their model size (total number of Gaussians). We find that performance of BIC improve 4.8% compare to those using the likelihood increment approach. We confirm that BIC is useful for mixture growing. It is also worth noting that using BIC produces the biggest model while using occupancy count produces the smallest.

6.2. Experiment II: Performance of tree pruning using BIC with multiple mixtures

In this set of experiments, we tested the effectiveness of pruning the state-tying tree. In the first experiment, we use BIC with a mixture of Gaussians representing the clustered-states, to prune the state-tying tree. The results are shown in Table 2. We find that the proposed method performs as good as the unmerged BIC method with a reduction of around 7% in model size. After we find the number of Gaussians in the first experiment, a control experiment using the exact number of Gaussians is performed. The model for this control experiment is created by removing the mixture components that contributed the least BIC gain. The result of the control experiment is shown in Table 3. Comparing the performance of the control experiment with the BIC merge with multiple mixtures, the merging consistently outperforms the control experiment by about 5.0%.

Method	feb89	oct89	feb91	size
BIC Method	96.25	95.42	96.70	9901
BIC Method+merge	96.45	95.53	96.60	9298

Table 2: Performance of Proposed Method

Baseline Exp.	feb89	oct89	feb91
BIC Method w merging	96.45	95.53	96.60
BIC Method w pruning	96.29	95.23	95.38

Table 3: Comparison between Different Pruning Methods

7. Discussion

In this paper, we have investigated the issue of state tying and mixture growing. In particular, we have proposed a new approach to handle the interaction between state-tying and mixture growing. We have shown that by merging triphone-states from the same parent using BIC with multiple mixtures results in a more that is more compact with better performance.

While the improvement is modest, we consider this a promising beginning. There are a number of possibilities that we plan to explore further, such as the tuning of λ. It was shown in [8] by tuning the complexity penalty λ better recognition performance can be obtained. In addition, we expect further improvements when BIC is also used in designing the state-tying tree and the tree is recursively merged.

8. Acknowledgment

This work is partially supported by the Hong Kong RGC under the grant number CA97/98.EG02.

9. REFERENCES

1. Young S. J., Odell J. J., Woodland P. C.,*Tree-Based State Tying for High Accuracy Acoustic Modelling*,ARPA Workshop on Human Language Technology,pp 286-291, 1994.
2. Young S. J., Woodland P. C.,*State Clustering in hidden Markov model-based continuous speech recognition*, Computer Speech and Language 8 pp.369-383
3. Normandin Y., *Optimal splitting of HMM gaussian mixture component with MMIE training*, Proceeding of ICASSP 1995 pp.449-552
4. Chen S. S., Eides E. M., Gales M. J. F., Gopinath R. A., Kanevsky D., Olsen P., *Recent improvement to IBM's speech recognition system for automatic transcription of broadcast news*, Proceeding of ICASSP 1999 pp.37-40.
5. Shinoda K., Watanabe T., *Acoustic modelling based on the MDL principle for speech recognition*, Proceeding of ICASSP 1997.
6. Bahl L. R., de Souza P. V., Gopalakrishnam P. S., Nahamoo D. Pichemy M. A., *Context Dependent Modelling of Phones in Continuos Speech Using Decision Trees*, Proc DARPA Speech and Natural Language Processing Workshop,pp. 264-270.
7. Chen S. S. *Clustering via the Bayesian information criterion with application speech recognition* Proceeding of ICASSP 1998.
8. Chou W. Reichl W..*Decision Tree State Tying Based on Penalized Bayesian Information Criterion*, Proceeding of ICASSP 1998
9. MaLaclan G. J., Basford K. E. *Mixture Models*, Marcel Dekker Inc,1988.
10. Young S. J., Odell J. J.,Olasen D. Valtchev V, Woodland P.,*The HTK Book(for HTK Version 2.2)* 1999
11. Schluter R., Machery W., Muller B. Ney H. *A Combined Maximum Mutual Informaiton and Maximum Likelihood Approach for Mixture Density Splitting*, Proceeding of ICASSP 1999.