3rd European Conference on
Speech Communication and Technology
EUROSPEECH'93
Berlin, Germany, September 19-23, 1993

THE USE OF STATE TYING IN CONTINUOUS SPEECH RECOGNITION

S.J. Young & P.C. Woodland

Cambridge University Engineering Department, England

ABSTRACT

This paper describes a method of robustly training context-dependent multiple Gaussian mixture HMM phone models without the need for *a posteriori* smoothing. The method involves clustering and then tying acoustically similar states within each allophone set in order to balance model complexity against the available data. The operational properties of the method are studied and results are presented for phone recognition on TIMIT. The method is shown to be robust, to give good recognition performance and to reduce computation in both recognition and training. All experiments were performed using the HTK portable HMM toolkit.

Keywords: HMM state clustering phone recognition TIMIT HTK

1. INTRODUCTION

Hidden Markov Models (HMMs) provide a sound basis for modelling both the inter- and intra-speaker variability of natural speech. However, to accurately model the distributions of real speech spectra, it is necessary to have quite complex output distributions. For example, in continuous density HMM systems, multiple Gaussian mixture components must be used to achieve good performance [6]. Furthermore, to deal with contextual effects such as coarticulation, context-dependent triphones are needed [4]. Thus, a speaker independent continuous speech HMM system will typically contain a large number of context-dependent models each of which contains a large number of parameters.

Unfortunately, the ability to arbitrarily increase model complexity is tempered in practice by the limited amount and the uneven spread of available training data. Thus, the key problem to be faced when building a HMM-based continuous speech recogniser is maintaining the balance between model complexity and available training data.

Traditional methods of dealing with this problem tend to be model-based. For example, for discrete (and tied-mixture systems) it is common to interpolate between triphones, biphones and monophones [4]. Recently, in an attempt to avoid the need for this *a posteriori* smoothing, both stochastic decision trees[1] and MAP estimation [2] approaches have been proposed. However, one of the limitations of model-based approaches is that the left and right contexts cannot be treated independently and since the distribution of training examples between left and right contexts will rarely be equal, this leads to a sub-optimal use of the data.

In this paper, a method of state tying based on the use of continuous density Gaussian mixture HMMs is described. This approach was first described at ICASSP '92 in the context of the generalised tying framework provided by the CUED HTK Toolkit [10]. It has since been refined and applied to more tasks. In particular, it was used to build the HTK recogniser used in the September 1992 DARPA Resource Management evaluation [8] and it will also be used for the forthcoming November 1993 ARPA Wall Street Journal evaluation.

State tying allows model complexity to be balanced against the amount of available data and as a result it enhances performance. Furthermore, state tying typically reduces the total number of states in a system by a factor of 5 thereby significantly reducing computation both in training and recognition. The method of state tying described here has some features in common with the senone system developed independently at CMU [3] for tied mixture systems.

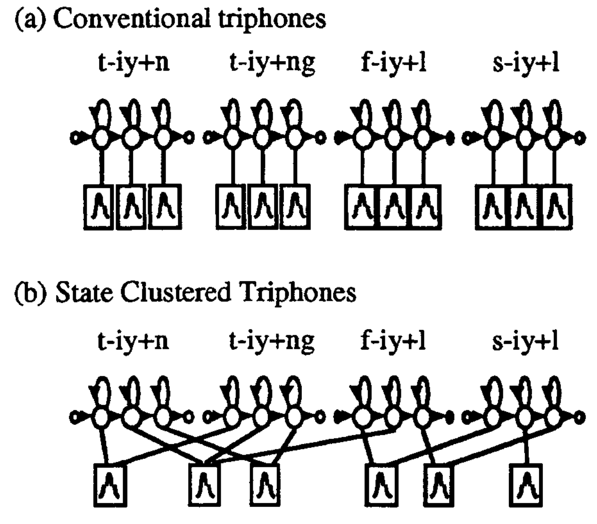
The remainder of this paper is organised as follows. In the next section, the algorithms used to build a tied-state HMM system are described. Following this, some properties of the algorithm are explored using phone recognition on the TIMIT database as the evaluation task. Finally, the results are discussed and some conclusions are presented. Performance results on the Resource Management database are given in a companion paper [9].

2. TIED STATE HMMS

2.1. Overview

Before discussing details, it is helpful to give an overview of the processes involved in the construction of a tied-state HMM system. The basic steps are as follows:

1. Create a set of single mixture monophone HMMs and train on data using context-independent transcriptions. Each such monophone represents a *base phone.*
2. For each required context-dependent model, clone the appropriate monophone and then retrain the


(a) Conventional triphones

(b) State Clustered Triphones

Figure 1. Illustration of State Tying

cloned models on data using context-dependent transcriptions. During this retraining, retain the state occupation counts of all models. The resulting context-dependent models represent *allophones* of the parent phone.

3. For each group of allophones which share the same parent, cluster and tie corresponding states. When the clustering is completed, check the total occupation count for each cluster of tied states. If it falls below an outlier threshold *RO* (typically 100), merge that cluster with its nearest neighbour.
4. For each group of allophones which share the same parent, compare corresponding states and merge all models which are identical. This step does not affect recognition accuracy but it does reduce the computational complexity of the decoder.
5. Successively increment the number of mixtures in each state and retrain the models. Terminate when some required number of mixture components is reached or when the performance on a development test set peaks.

The end result is a set of context-dependent models in which there is typically a high degree of state tying. Figure 1 illustrates this where the naming convention `A-B+C` denotes the allophone of phone `B` occurring with a left context of `A` and a right context of `C`. Part (a) of this figure shows the original untied triphones and part (b) illustrates the sharing of state distributions on completion of the clustering and tying process. Since each tied state has a guaranteed minimum number of occurrences in the training data, it can be robustly trained. Notice that a key assumption in this procedure is that there is sufficient training data to obtain reasonable estimates for the initial untied single mixture context-dependent triphone set. In practice, just one example of each triphone is sufficient since the variance can be clamped or backed-off to the corresponding monophone variance.

2.2. Clustering

Given a set of all the corresponding states of the allophones of some base phone, a set of clusters are formed using the following furthest neighbour hierarchical clustering algorithm

```
create 1 cluster for each state;
find i and j for which g(i,j) is minimum;
while (g(i,j) < TC) {
   merge clusters i and j;
   find i and j for which g(i,j) is minimum;
}
```

where *TC* is a threshold which determines the maximum size of any cluster and `g(i,j)` is the inter-group distance between clusters `i` and `j` defined as the maximum distance between any state in cluster `i` and any state in cluster `j`. It may be noted that a k-means top-down clustering algorithm has also been tried but it did not work as well, the main difficulty being that it tended to produce clusters which were not of uniform size.

The inter-state distance is the square root of the divergence between the two Gaussian pdfs. For diagonal covariances, this is given by

$$d(i,j) = \left[\frac{1}{V}\sum_{k=1}^{V}\frac{\sigma_{ik}^2}{\sigma_{jk}^2}+\frac{\sigma_{jk}^2}{\sigma_{ik}^2}-2 + \left(\frac{1}{\sigma_{ik}^2}+\frac{1}{\sigma_{jk}^2}\right)(\mu_{ik}-\mu_{jk})^2\right]^{\frac{1}{2}} \quad (1)$$

where μ_i is the mean for state i, σ_i^2 is the variance for state i, and V is the dimensionality of the input data. We have also tried using the following related but simpler distance metric.

$$d(i,j) = \left[\frac{1}{V}\sum_{k=1}^{V}\frac{(\mu_{ik}-\mu_{jk})^2}{\sqrt{\sigma_{ik}^2\sigma_{jk}^2}}\right]^{\frac{1}{2}}. \quad (2)$$

This has been found to work equally well and is, in fact, more robust with sparse data since it not as sensitive to differences in the variances.

2.3. Mixture Incrementing

The final stage in the construction of the tied-state HMM system is to convert the single Gaussian mixture output distributions to multiple mixtures. This is done in stages, incrementing all shared pdf's by 1 or 2 mixtures at each stage. Given m mixtures, a pdf is converted to $m+1$ mixtures by finding the mixture component with the largest weight. The weight of this mixture component is halved and then the mixture is cloned. The two identical mean vectors are then perturbed by adding 0.2 standard deviations to one and subtracting the same amount from the other. After each stage of mixture splitting, all of the HMM parameters are updated by performing two iterations of embedded Baum-Welch re-estimation.

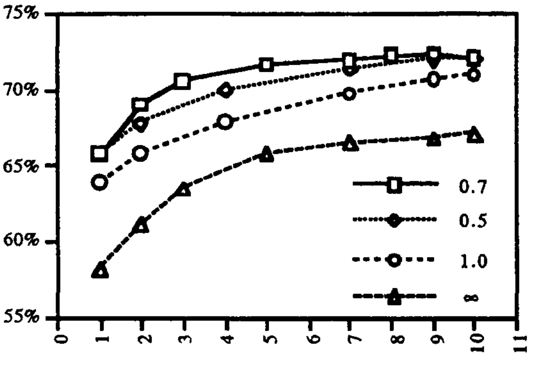


Figure 2. Recognition Accuracy vs Number of Mixtures for 4 Different Cluster Thresholds *TC*

For a target system of M mixtures, this method of building multiple-mixture systems has been compared with the more conventional approach of using a segmental k-means procedure to initialise the required M mixtures and then retraining using embedded Baum-Welch re-estimation. The results were similar for both approaches, however, the iterative mixture splitting approach has the advantage that the number of mixtures can be continuously increased until a peak in performance is reached.

3. EXPERIMENTAL EVALUATION

This section examines experimentally the properties of the state-clustering algorithm for different choices of cluster threshold *TC* and outlier threshold *RO*. The task chosen for this study is phone recognition on the TIMIT database. This task was selected because it is relatively simple to build and test models, and yet performance is closely correlated to that achieved on more complex word based tasks such as Resource Management. For similar reasons, biphones were used in preference to triphones to keep the computational complexity manageable whilst achieving reasonable levels of performance.

The experimental conditions were the same as those used by Lee in his baseline TIMIT experiments [5]. The 61 phone TIMIT label set was mapped down to a 48 set and three state left-right HMMs were used. These were cloned to give 1748 right context-dependent biphones. The data parameterisation consisted of 12 MFCC coefficients and normalised energy plus 1st and 2nd order differences. The models were trained on all *si* and *sx* sentences in the TIMIT training corpus making 3694 sentences in total. Recognition was based on a 300 sentence subset of the TIMIT test corpus and it used a standard Viterbi decoder with a context-independent bigram estimated from the training data transcriptions. All results are stated in terms of accuracy and for scoring purposes only, the 48 phone set was folded into a reduced 39 phone set as in Lee. The bigram scaling factor was set to be constant (at 8.0) and typically the percentage correct was around 5% greater than the accuracy due to insertion errors.

Figure 2 shows the increase in phone recognition accuracy as the number of mixtures are incremented. As can be seen, the choice of cluster threshold is not critical. The general trend is that increasing the cluster threshold leads to fewer states and therefore more data per state. This allows a greater number of mixtures to be reliably estimated and performance increases. However, as *TC* gets large, the system tends towards a monophone system and fails to model important context dependencies in the data. When this happens performance starts to drop. In each clustering case, the outlier threshold was held constant at 100.

TC	*#HMMs*	*#states*	*#mixes*	*#comps*
∞	48	144	7	1008
1.0	900	754	2	1508
0.7	1176	1142	1	1142
0.5	1349	1617	1	1617
0	1748	5244	2	10488

Table 1. Number of distinct HMMs, states, mixtures per state, and mixture components for differing values of cluster threshold *TC*, in each case the number of mixtures is that which gives an accuracy of approx. 66%

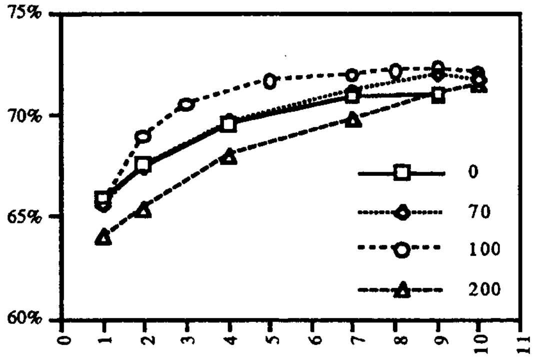


Figure 3. Recognition Accuracy vs Number of Mixtures for 4 Different Outlier Thresholds *RO*

Table 1 illustrates the effect of varying the clustering threshold further by showing the number of distinct models, states, and mixture components as *TC* is decreased from ∞ which corresponds to monophones down to 0 which corresponds to completely untied triphones. In each case, the number of mixture components per state is chosen to give an accuracy of approx. 66%. Notice that the value of *TC* which gives the best overall performance (0.7) is also the value which gives the minimum total number of mixture components for any of the triphone systems. Furthermore, this total number of mixture components is similar to that required by the multiple-mixture monophone system to achieve similar performance. The case of *TC* equal to zero corresponds to the simple triphone case. It may be noted that there is insufficient data to train more than 2 mixture components and hence this represents the maximum performance that can be achieved without state clustering. Notice also that the reduction in the number of models achieved by the state clustering is much less than the reduction in the number of states.

Figure 3 shows the effect of varying the outlier

Clusters for state 4 of phoneme /k/	
iy	y
aa, ah, ay	n, en, l, w
oy, el, ao, ow	ng, ih, ix, eh, ae, ey
er, r	epi, vcl, cl, th, dh, sil
uw, uh	m, ax, t, zh
aw	v, f, z, hh, sh, s
12 Clusters total	

Figure 4. Right Acoustic Contexts of /k/

threshold *RO*. Again the precise setting is not critical but the importance of having an adequate threshold is clearly seen. When *RO* is zero, some of the context-dependent models have an inadequate amount of training data and under-training becomes a problem. This is reflected in lower accuracy even though these models have more parameters. When *RO* is too high, acoustically distinct contexts are merged unnecessarily and again the performance drops.

To gain an impression of the type of acoustic contexts which are merged in the clustering process, Fig 4 shows the clusters formed for state 4 (i.e. the right-most emitting state) of the allophones of the voiceless stop /k/ for values of $TC = 0.7$ and $RO = 100$. As can be seen most of the groupings appear to be linguistically reasonable.

4. IMPLEMENTATION

All of the above experiments were performed and can be reproduced using the *HTK* portable HMM toolkit [11]. This toolkit contains around 20 tools of which 4 were predominant in the experiments described here. The state clustering, the mixture incrementing and the monophone to biphone cloning were implemented as edit commands to a HMM definition editor called `HHEd`. Embedded Baum-Welch re-estimation was performed by a tool called `HERest` whose use was interleaved with applications of `HHEd`. Finally, performance evaluation used a Viterbi recognition tool called `HVite` followed by a DP-based string matching results analysis tool called `HResults`.

5. DISCUSSION AND CONCLUSIONS

In this paper, a method of building context-dependent multiple mixture component HMMs has been described which balances model complexity against the amount of available training data. States are clustered in two stages. Firstly, similar acoustic states are merged and then secondly, any cluster with insufficient training data is merged with its nearest neighbour. Although each of these stages requires a corresponding threshold to be set, the results presented show that the setting of these thresholds is not critical.

Once the clusters have been formed, the number of mixture components in each state can be increased uniformly across all states. This has been shown to lead to very accurate models. The best performance achieved for phone recognition using right-context dependent biphones on the TIMIT database was 76.7% correct and 72.3% accuracy. These results are comparable with the best reported elsewhere for HMM systems although they are slightly below those reported by Robinson of 78.6% correct and 75.0% accuracy for his recurrent neural net recogniser[7].

Overall, tied-state continuous density HMM systems have proved to be easy to build and to give good performance over a wide range of tasks. The method has been used successfully for both phone recognition as described here and for 1000 word recognition as described in a companion paper[9]. Current work is applying the technique to 20,000 word recognition on the Wall Street Journal Task.

REFERENCES

[1] Bahl LR, de Souza PV, Gopalakrishnan PS, Nahamoo D, Picheny MA. *Context Dependent Modeling of Phones in Continuous Speech Using Decision Trees.* Proc DARPA Speech and Natural Language Processing Workshop, pp264-270, Pacific Grove, Calif, Feb, 1991

[2] Gauvain L, Lee C-H. *Bayesian Learning for Hidden Markov Model with Gaussian Mixture State Observation Densities.* Speech Communication, Volume 11, Nos 2-3, (Eurospeech '91), pp205-214, 1992

[3] Hwang M-Y, Huang X. *Subphonetic Modeling with Markov States - Senone.* Proc ICASSP, Vol 1, pp33-36, San Francisco, 1992

[4] Lee K-F. *Context-Dependent Phonetic Hidden Markov Models for Speaker Independent Continuous Speech Recognition.* IEEE Trans ASSP, Vol 38, No 4, pp599-609, 1990

[5] Lee K-F, Hon H-W. *Speaker Independent Phone Recognition Using Hidden Markov Models.* IEEE Trans ASSP, Vol 37, No 11, pp1641-1648, 1989

[6] Rabiner LR, Juang B-H, Levinson SE, Sondhi MM. *Recognition of Isolated Digits Using HMMs with Continuous Mixture Densities.* AT&T Technical J, Vol 64, No 6, pp1211-1233, 1985

[7] Robinson AJ. *Several Improvements to a Recurrent Error Propagation Network Phone Recognition System.* Cambridge University Engineering Dept, Technical Report, CUED/F-INFENG/TR.82, 1991

[8] Woodland PC, Young SJ. *Benchmark DARPA RM Results with the HTK Portable HMM Toolkit.* Proc DARPA Continuous Speech Recognition Workshop, Stanford, Sept, 1992

[9] Woodland PC, Young SJ. *The HTK Tied State Continuous Speech Recogniser.* Proc Eurospeech, Berlin, Sept, 1993

[10] Young SJ. *The General Use of Tying in Phoneme-Based HMM Speech Recognisers.* Proc ICASSP, S66.5, San Francisco, March, 1992

[11] Young SJ. *HTK Version 1.4: User, Reference and Programmer Manual.* Cambridge University Engineering Dept, Speech Group, August, 1992