Restructuring HMM States for Speaker Adaptation in Mandarin Speech Recognition

Xianghua Xu, Jie Zhu and Qiang Guo

Department of Electronic Engineering
Shanghai Jiaotong University, Shanghai, China
{xuxianghua_sd,jiezhu,guoqiang_sjtu}@sjtu.edu.cn

Abstract
With the tendency of posterior probability taken into account, a state-restructuring method is proposed based on confusions between HMM states. In the method, HMM state is restructured by sharing Gaussian components with its related states and the re-estimation of the increased-parameters, i.e., the inter-state weights, is derived under the EM framework. Experiments are performed on speaker-independent large vocabulary continuous Mandarin speech recognition. The results show the state-restructured systems outperform the baseline system and the combining with MLLR adaptation can lead to consistent and significant improvement on recognition accuracy over MLLR.

1. Introduction
Continuous density hidden Markov models (HMMs) with phonetic decision tree based state-tying are widely used in large vocabulary continuous speech recognition (LVCSR) systems. As almost all LVCSR systems are speaker-independent, speaker adaptation techniques have been applied to improve recognition performance by doing model parameter estimation or transformation to fit the characteristics of the testing speech. Commonly used adaptation algorithms include Maximum Likelihood Linear Regression (MLLR) and Maximum a Posterior (MAP) estimation. Although these methods are useful, they don’t change the structure of decision tree or the structure of tied-states [1]. Individual states of the generic model may be separated or tied together, or Gaussian components within states may be split or not according to the likelihood estimation and occupation counts from the training data, which causes that such structures may not reflect specific features in the testing data. This situation can become acute when distributions of training data and adaptation data are substantially different.

In recent study Luo [2] proposes the probabilistic classification of HMM states. The improvement reported in [2] suggests the importance of having a proper state structure. This study is related to what we are proposing here in the state-restructuring method. The method is by using the weighted Gaussian components of some states which are commonly confused with the hypothesized state, thus detecting such confusions. The set of the commonly confused states is constructed from the N-best list for the hypothesized state. The advantage of this method over [2] is that this method has the tendency of improving the posterior probability. Particularly, it requires no re-estimation for original model parameters.

In this paper, we firstly investigate to restructure states of speaker-independent acoustic models by using a small amount of adaptation data. The aim is to increase posterior probability to adaptation speech. Secondly, we extend the method to using training data. Thus, each state is inflated by sharing weighted components with some related states. The final state-restructured HMMs are transformed with MLLR adaptation method. The rest of the paper is organized as follows. In section 2 the proposed method is described. In section 3, experimental results are provided. A conclusion is made in section 4.

2. State-restructuring method
Assume an initial HMM set is yielded with conventional algorithms and let Ω be the set of HMM states. Figure 1 illustrates the process of state-restructuring and speaker adaptation.

2.1. Restructuring states with adaptation speech
When recognize adaptation speech with speaker-independent models, many states are confused with others, e.g., state A is recognized to state B (B=\{A\}) by error. Adaptation speech has similar characteristics with testing speech, same errors may occur for testing. Thus restructuring states with adaptation speech can directly result in word error rate reduction. From adaptation context: \(X_t=[X_1,...,X_n]\), we build the second state set of Φ, which includes all states that occur in adaptation data. As for context \(X_t\), the corresponding feature vector is \(O=(o_1,...,o_T)\) and the aligned state sequence is \(S_t=[s_1,...,s_T]\) (\(S\subseteq\Omega\)). By Viterbi decoding algorithm [3], we get the other state sequence of \(R_t=[r_1,...,r_T]\) \((\Omega\subseteq\Omega)\). Define \(S\) as the actual state sequence and \(R\) the recognized state sequence. Compare the two sequences, we get states \(s_t\) and \(r_t\) for observation feature vector \(o_t\). If \(s_t\neq r_t\), call \(r_t\) the related state of \(s_t\), and define confusion \(C_{s_t|r_t}\) as

\[
C_{s_t|r_t} = \frac{P(o_t|s_t)}{P(o_t|r_t)} \quad (1)
\]

Since state \(s_t\) is recognized to \(r_t\) by error, if \(s_t\neq r_t\), \(P(o_t|s_t) > P(o_t|r_t)\), i.e., \(C_{s_t|r_t} > 1\). From definition (1), it can be seen that the larger \(C_{s_t|r_t}\) is, the more possible it is that \(s_t\) is recognized to \(r_t\). Thus, if \(r_t\) is shared with \(s_t\), i.e., share...
the weighted Gaussian components in \( r_i \) with \( s_i \) and accordingly change the structure of \( s_i \), the posterior probability \( P(o|s_i) \) will increase, therefore, the error reduction can be achieved.

For arbitrary state \( s \notin \Phi \), define its related state set of \( R' \) from the recognition results of the adaptation speech. Each state in \( R' \) is the related state to \( s \). Restructure \( s \) with \( r (r \in R') \), the final Gaussian mixture function is

\[
b(s) = \sum_{r \in R'} w_{sr} P(r) + w_0 P(s) = \sum_{r \in R'} w_{sr} P(r)
\]

where \( R' = R \cup \{s\} \) and \( w_{sr} = w_0 \). Initialize \( w_0 \) to \( 1-D \), where \( D \) is a hand-set constant. The initial weights \( w_{sr} \) and the probability function \( P(r) \) are

\[
w_{sr} = \begin{cases} \frac{C_{r}}{D}, & \text{for } r = s \\ 1-D, & \text{for } r \neq s \end{cases}
\]

where \( K \) is the number of Gaussian mixtures, \( N(\mu_{r,k}, \delta_{r,k}) \) represents multi-Gaussian function, \( m_{r,k}, \mu_{r,k} \) and \( \delta_{r,k} \) are the mixture weight, mean vector and dialogue covariance matrix of the \( k \) th mixture component. Accordingly, the restructured-state has two level weights: the inter-state weight \( w_{sr} \) and the intra-state weight \( m_{r,k} \). They satisfy

- **Intra-state weight**: \( \sum_{k=1}^{K} m_{r,k} = 1, 0 \leq m_{r,k} \leq 1 \).
- **Inter-state weight**: \( \sum_{r \in R'} w_{sr} = 1, 0 \leq w_{sr} \leq 1 \).

Then, inter-state weights are re-estimated with the maximum likelihood criterion. The objective function is given by the log-likelihood function as

\[
L(O_r) = \sum_{o \in O_r} \log \left( \sum_{r \in R'} w_{sr} P(o|r) \right)
\]

where \( O_r \) represents the set of observation feature vectors that belong to state \( s \) in adaptation data. Our goal here is to find \( w_{sr} \) which maximizes the upper log-likelihood function. However, since there is no direct solution for the maximization, an EM algorithm [4] is suggested. The EM algorithm requires an auxiliary function as

\[
Q\left( w_{sr}, \overline{w_{sr}} \right) = \mathbb{E} \left[ \log P\left( O_r, s \bigg| \overline{w_{sr}} \right) O_r, \overline{w_{sr}} \right]
\]

We can update the weight \( w_{sr} \) by equating the gradient of (6) with respect to \( \overline{w_{sr}} \) to zero under the constraint of \( \sum_{r \in R'} w_{sr} = 1 \). Then we get

\[
\overline{w_{sr}} = \frac{\sum_{r \in O_r} \sum_{k=1}^{K} \gamma(s, r, k) m_{r,k} N(o|\mu_{r,k}, \delta_{r,k})}{\sum_{r \in O_r} \sum_{k=1}^{K} \gamma(s, r, k)}
\]

Here, the function

\[
\gamma(s, r, k) = \frac{w_{rs} m_{r,k} N(o|\mu_{r,k}, \delta_{r,k})}{\sum_{r \in R'} \sum_{k=1}^{K} w_{rs} m_{r,k} N(o|\mu_{r,k}, \delta_{r,k})}
\]

probability of occupying the \( k \) th component of state \( r \) for observation \( o (o \in O_r) \). Let \( \tilde{s} \) denote the original state before state-restructuring, the original log-likelihood in state \( s \) is \( L(O_r) = \sum_{o \in O_r} \log P(o|\tilde{s}) \). Compute the term

\[
\Delta L(O_r) = L(O_r) - L(O_r)
\]

as log-likelihood increasing after state-restructuring. The average increment in adaptation data is

\[
\Delta L = \frac{1}{\text{size} (\Phi)} \sum_{o \in O_r} \Delta L(O_r)
\]

Then we take \( \Delta L \) as the threshold to be applied in the following section.

### 2.2. Restructuring states with training speech

As the adaptation speech is limited, large number of states may be left unchanged after the upper state-restructuring with only adaptation data. Thus, the improving to system performance may be slight. Since training speech is from a large number of speakers, it can cover the acoustic space to some extent. Therefore, we suppose states, that tend to be confused with some states in training set, still tend to be confused to these states in test set. Based on the hypothesis, we extend the state-restructuring method to using training data.

Define the third state set of \( \Psi (\Psi = \Omega \cup \Phi) \), which include all the other states that don’t occur in adaptation data. State-restructuring with training speech is implemented as following steps.

1. **(Step-1)** For training contexts \( Y = \{Y_1, \ldots, Y_n, \ldots\} \), align states and training context \( Y \), by Viterbi decoding and get the most likely recognized state sequence \( \eta_{\text{i}} \).

2. **(Step-2)** As for context \( Y \), construct a network: LM. Based on the network LM, Viterbi alignment method is used to give the actual state sequence \( \gamma_i \).

3. **(Step-3)** Do Step-1 and Step-2 for all training contexts.

4. **(Step-4)** For arbitrary \( s (s \in \Psi) \), select the related state set \( R' \) from \( \eta_{\text{i}} \) by comparing the two sequences of \( \eta_{\text{i}} \) and \( \gamma_i \). Compute confusion \( C_{dr} (r \in R') \). According to \( C_{dr} \), sort states in \( R' \) by descending. Let the size of \( R' \) be \( I \).

5. **(Step-5)** Restructure state \( s \) as illustrated in Figure 2: Use the top \( i \) (\( i \in \mathbb{N} \)) states to restructure \( s \) and compute the corresponding log-likelihood increasing \( \Delta L_s \) as described in section 2.1. If \( \Delta L_s > \Delta L \), stop the process, else, use the \( i+1 \) top states until \( \Delta L_s > \Delta L \) with \( I \leq I' \). The final \( s \) is restructured with the \( i \)-best states or left unchanged.
Do Step-4 and Step-5 for all states in $\Psi$. 

Training data

Adaptation data

Original SI HMMs

State-restructuring

Speaker adaptation with MLLR

SA HMMs

Figure 1: Process of state-restructuring and speaker adaptation

Initialization $i=0$

$i=i+1$

$i > I_s$

No

Use the top $i$ states to restructure $s$ and compute $\Delta L_i$

$\Delta L_i > \Delta_0$

Yes

Unchanged state $s$

Restructured state $s$ with the $i$ related states

3. Experiments

The proposed state-restructuring method for speaker adaptation is evaluated on the LVCSR Mandarin dictation task. The database is from the National 863 High-Tech Project. The training set contains 36,201 sentences from 68 female speakers. Another 14 female speakers build up the testing set. Each has 40 adaptation sentences and 30 testing sentences. MLLR speaker adaptation with multiple transforms is used in following experiments. Feature vector has 39 elements, consisting of 12 MFCCs and the normalized energy plus their 1st and 2nd order derivatives. The system uses context dependent tri-phone units for modeling Mandarin tonal syllables, which include 181 basic phonemes: 27 initial, 153 tonal final and a silence model. All the models have three emitting states, with a strictly left-to-right topology and 16 Gaussian components per state. To illustrate the effectiveness of the proposed algorithm, only the acoustic performances are given. The baseline system was trained by HTK 3.1[5] and a Word (tonal syllable) Recognition Rate (WRR) of 73.22% was achieved by using 3021 tied-states after decision-tree based state tying [6].

The experimental conditions include baseline speaker-independent recognition (SI), the state-restructured system by adaptation data (SR1), state-restructured system by both adaptation data and training data (SR2), MLLR adaptation for the baseline (MLLR), MLLR adaptation for SR1 system (SR1/MLLR), and MLLR adaptation for SR2 system (SR2/MLLR).

Table 1 lists the WRR comparison of SR1 and SR2 with different number of adaptation sentences. The first observation is that both system performances keep increasing as the amount of adaptation data increases, which proves the viability of restructuring states with adaptation data. The second observation is that from a small amount of adaptation data, only a slight improvement in accuracy can be obtained in SR1, whereas the improvement is noticeable in SR2. For instance, from only 1 adaptation sentence, compared with the baseline system, the SR2 system improves WRR by 1.30%, while the SR1 has almost no performance increasing. Thus, we can owe the performance increasing in SR2 to the state-restructuring with training data, implying the hypothesis made in section 2.2 is reasonable. Particularly, when the adaptation data is very limited, state-restructuring with training data can take more advantage of priori knowledge.

Table 1: WRR comparison of SR1 and SR2 with different number of adaptation sentences

<table>
<thead>
<tr>
<th>Adaptation sentences</th>
<th>SR1</th>
<th>SR2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>73.23%</td>
<td>74.17%</td>
</tr>
<tr>
<td>2</td>
<td>73.27%</td>
<td>74.23%</td>
</tr>
<tr>
<td>3</td>
<td>73.33%</td>
<td>74.31%</td>
</tr>
<tr>
<td>5</td>
<td>73.41%</td>
<td>74.43%</td>
</tr>
<tr>
<td>10</td>
<td>73.59%</td>
<td>74.60%</td>
</tr>
<tr>
<td>20</td>
<td>73.83%</td>
<td>74.67%</td>
</tr>
<tr>
<td>30</td>
<td>74.01%</td>
<td>74.83%</td>
</tr>
<tr>
<td>40</td>
<td>74.25%</td>
<td>75.09%</td>
</tr>
</tbody>
</table>

To evaluate the state-restructured models after speaker adaptation, recognition results in conditions of SI, MLLR, SR1/MLLR and SR2/MLLR are summarized in Figure 3 for the varying number of adaptation sentences. As is illustrated by Figure 3, compare with the baseline system and MLLR alone, a significant error reduction is achieved by using state-restructured models as the amount of adaptation data increases. This is not very surprising since as more adaptation data became available, the state-restructured model can make better use of adaptation data, which greatly benefits the following adaptation. Besides, the state-restructured model is larger than the baseline and a larger model is likely to match test data better after adaptation than a smaller model [7].
4. Conclusions

In this paper we have proposed a new state-restructuring method based on confusions between states, in which a state is restructured by sharing Gaussian components with its related states. Experimental results indicate that the model restructured by adaptation speech outperforms the baseline. And the model restructured by both adaptation speech and training speech works reasonably well with small amounts of adaptation data. Either state-restructured model can lead to consistent and significant improvements to MLLR. As future work, this method is being extended to other applications such as unsupervised adaptation.

5. References


