Mean and Covariance Adaptation Based on Minimum Classification Error
Linear Regression for Continuous Density HMMs

Haibin Liu, Zhenyang Wu
Dept. of Radio Engineering, Southeast Univ., Nanjing, 210096, China
tiger_lhb@seu.edu.cn

Abstract

The performance of speech recognition system will be significantly deteriorated because of the mismatches between training and testing conditions. This paper addresses the problem and proposes an algorithm to adapt the mean and covariance of HMM simultaneously within the minimum classification error linear regression (MCELR) framework. Rather than estimating the transformation parameters using maximum likelihood estimation (MLE) or maximum a posteriori, we proposed to use minimum classification error (MCE) as the estimation criterion. The proposed algorithm, called IMCELR (Improved MCELR), has been evaluated on a Chinese digit recognition tasks based on continuous density HMM. The experiments show that the proposed algorithm is more efficient than maximum likelihood linear regression with the same amount of adaptation data.

1. Introduction

Current state-of-the-art speech recognition systems are capable of achieving good performance in the well-defined environments. However, under the practical environments, the system performance will dramatically degrade because of the existing mismatch between the training and testing data, and for complex speech recognition systems a large amount of data is required to retrain the system for a new acoustic environment. Hence, it is very desirable to be able to improve the performance of an existing system while only using a small amount of environment-specific adaptation data. In the past 20 years, research on model adaptation algorithm has become an active research area, and so far, many model adaptation algorithms based on sparse data were presented. The important algorithm in this area is the maximum likelihood linear regression, presented by Leggetter & Woodland [1]. The MLLR algorithm estimates a set of linear transformations for the mean parameters of a mixture-Gaussian HMM system to maximize the likelihood of the adaptation data. It should be noted that while MLLR was initially presented for speaker adaptation, since it reduces the mismatch between a set of models and adaptation data, so it also can be used to perform environmental adaptation by reducing mismatch due to channel or additive noise effects. Other than the theoretical framework of the MLLR, many researchers also have great interesting in MCE adaptation of parameters of mixture-Gaussian HMM. MCE was first thoroughly reported in [2]. Instead of assuming that the parametric model used in speech recognition characterizes the true distribution of the data, MCE approach is discriminative function based pattern recognition. But it is well known that direct MCE adaptation works well when sufficient (w.r.t the number of parameters being adapted) amount of adaptation data are available. However, with only small amount of adaptation data, the direct MCE adaptation for HMM does not work so well. So the adaptation approach based on MCE used for small amount adaptation data set was studied. In [3], Xiaodong He and Wu Chou presented a linear transformation adaptation based on minimum classification error. In the presented algorithm, it only adjusted the mean parameters of HMM while the covariance is invariant, and in the next paper [4], they again present an algorithm to adjust the variance parameters and fixed the mean of the HMM, and demonstrated that both approach are effective. Because the mean and variance were adapted respectively, so the performance did not reach the optimality.

In this paper, we proposed a modified algorithm to adjust the mean and variance parameters of HMM simultaneously based on the MCELR to minimize the loss function. This approach has a monotonic loss minimization property as presented by Xiaodong He [3], which is critical for model adaptation when only a small amount of adaptation data is available, but lots of parameters to estimate. Because we adapt the mean and covariance parameters of Gaussian component at the same time, the optimality can be expected and the mismatch between the training and test data can be reduced at high limit.

2. Algorithm Analysis

2.1. Minimum Classification Error Using String Model

MCE is based on minimizing the recognition error rate on the training data. In the MCE framework formulated in [5], a sigmoid function is adopted to approximate the
empirical classification error rate for the super string \( X \) under current model parameter set \( \Lambda \) as follows:

\[
L_c(X; \Lambda) = \frac{1}{1 + \exp(-\alpha d_c(X, \Lambda) + \beta)}.
\]  

(1)

Where \( X \) denotes the training data, which is constructed by concatenating the limited adaptation utterance into one string as in [3].

In the aforementioned equation, the values of constant \( \alpha \) and \( \beta \) control the slope and mid point of the sigmoid function. When \( \alpha \to \infty \), the loss function, \( L_c(X; \Lambda) \), is infinitesimally close to the true misclassification error. For tractable processing, we take \( \alpha = 1 \), \( \beta = 0 \) and \( d_c(X, \Lambda) \) is the misclassification measure defined as follows:

\[
d_c(X; \Lambda) = -g_c(x_i, W_i; \Lambda) + \left[ \frac{1}{V-1} \sum_{i \neq j} g_j(x_i, W_i; \Lambda)^\eta \right]^{1/\eta} \]  

(2)

Where \( V \) denotes the class number, and \( W_i \) is the correct transcript lexical word string, and \( W_i \) is the confusing word string that are different \( W_i \). \( \eta \) is a positive number. If we let \( \eta \) approach infinite, then equation (2) becomes

\[
d_c(X; \Lambda) = -g_c(x_i, W_i; \Lambda) + g_s(x_i, W_i; \Lambda), \]

(3)

where \( s = \max_{s \neq j} g_j(x_i, W_j; \Lambda) \), and \( g(X, W; \Lambda) \) is a discriminative function for recognition decision-making. For an HMM-based speech recognizer that uses Viterbi decoding, the discriminant function \( g_c(x_i, W_i; \Lambda) \) can be the log-likelihood score as follows:

\[
g_c(x_i, W_i; \Lambda) = \log(f(x_i, | W_i; \Lambda)). \]

(4)

Substituting equation (3) and (4) into equation (1), the following equation can be got,

\[
L_c(X; \Lambda) = \frac{f(x, W_i; \Lambda)}{f(x, W_i; \Lambda) + f(x, W_i; \Lambda)}. \]

(5)

In conventional MCE algorithm, minimizing the equation (5) is implemented by GPD algorithm. It is well known that it is very difficult in selecting the step size, and the step size has a critical impact on algorithm performance. In order to improve the performance of the algorithm and avoid this problem, we adopt the algorithm presented by [3] to minimize the equation (5).

As in [3], It is not direct to minimize the \( L_c(X, \Lambda) \), but consider

\[
P(\Lambda) = 1 - L_c(X, \Lambda). \]

(6)

It is obvious that minimizing \( L_c(X, \Lambda) \) is equivalent to maximize

\[
P(\Lambda) = \frac{f(X, W_i; \Lambda)}{f(X, W_i; \Lambda) + f(X, W_i; \Lambda)}. \]

(7)

And give the EM formulation to maximize it by iteration as follows

\[
Q(\Lambda|\Lambda') = \sum_{t,m} \Delta \gamma(t,m) \log(f(x_t, s_t = m|\Lambda) + \\
\sum_{t,m} d'(t,m) f(x_t, s_t = m|\Lambda') \log(f(x_t, s_t = m|\Lambda) \]  

(8)

Where \( \gamma(t, m) \) denotes a space with \( P \times T \) dimensions, and \( P \) is the feature dimension \( T \) is the total frame number of the training set \( X \). \( \Lambda \) denotes the current model parameters set and \( \Lambda' \) stands for the model parameters set of the last iteration. And \( \Delta \gamma(t,m) \) can be calculated by the following equation

\[
\Delta \gamma(t,m) = \frac{f(X, W_t; \Lambda') \gamma(t, m, W_t) - \gamma(t, m, W_t)}{f(X, W_t; \Lambda') + f(X, W_t; \Lambda')}. \]

(9)

Where \( \gamma(t, m, W) = p(s_t = m \mid X, W; \Lambda') \) is the a posteriori probability of occupying the Gaussian component \( m \) at time \( t \), given the data \( X \) and the reference word string \( W \), and \( d'(t,m) \) can be computed by the following equation

\[
d'(t,m) = \sum_{t,m} d(s) / f(X, W_t; \Lambda'). \]

(10)

Where \( d(s) \) denotes a positive constant and it’s calculation equation will be given later.

2.2. Mean and Covariance Adaptation by IMCEL

In the framework of HMMs, the model’s parameters can be characterize as \( \Lambda = \{ \lambda_i \}_{1 \leq i \leq V} \), and \( \lambda_i = \{ \mu^{(i)}_t, A^{(i)}, B^{(i)} \} \), where the superscript \( i \) denotes the class index, \( a^{(i)} \) and \( A^{(i)} \) denote the initial probability and transmitting probability respectively. During the adaptation, it can be considered they are fixed, and \( B^{(i)} \) denotes the observation sequence probabilities of each Gaussian component of each state for the HMM. In continuous density HMMs with mixture Gaussian densities, the Gaussian component is characterized by its mean and covariance matrix and denoted generally as:

\[
f(x_t | \mu_\mu, \Sigma_m, \lambda_i) = N(x_t | \mu^{(i)}_m, \Sigma^{(i)}_m) \]  

(11)

\[
= \left( 2\pi \right)^{-D/2} \left| \Sigma^{(i)}_m \right|^{-1/2} \exp \left\{ -\frac{(x_t - \mu^{(i)}_m)^T \Sigma^{(i)}_m (x_t - \mu^{(i)}_m)}{2} \right\}. \]

In most of linear regression based adaptation approaches, all of the Gaussian components from all continuous density HMMs will be clustered into several classes. In each class, the same linear transformation can be applied for all Gaussian components. In this paper, in order to get the close form solution, we take the linear transformation for the mean and covariance as follows, and omit the superscript \( i \):
\[ \hat{\mu}_m = \mu_m + \Sigma_m^{1/2} b \]
\[ \hat{\Sigma}_m = \Sigma_m^{1/2} A (\Sigma_m^{1/2})^T. \]

Substituting equation (12) into equation (11) and applied log operation for the two sides, then get the following equation
\[ q(x_i | \hat{\mu}_m, \hat{\Sigma}_m; \Lambda) = \log f(x_i | \hat{\mu}_m, \hat{\Sigma}_m; \Lambda) = -\frac{1}{2} \log |\Lambda| - \frac{1}{2} [\Sigma_m^{-1/2} (x_i - \mu_m) - b]^T A^{-1} [\Sigma_m^{-1/2} (x_i - \mu_m) - b] + C \]

(13)

Where C denotes a constant that is independent of A and b.

Substituting equation (12) and (13) into equation (8), then get the following equation,
\[ Q(\Lambda | \Lambda') = \sum_{i,m} [\Delta \gamma(t,m)] q(x_i | \hat{\mu}_m, \hat{\Sigma}_m; \Lambda) + \sum_{i,m} d'(t,m) f(x_i | s_t = m; \Lambda') q(x_i | \hat{\mu}_m, \hat{\Sigma}_m; \Lambda) dx_i \]

(14)

In order to derive the iteration equation of the linear transformation parameters, A and b, we can respectively set
\[ \frac{\partial Q(\Lambda | \Lambda')}{\partial b} = 0, \]
\[ \frac{\partial Q(\Lambda | \Lambda')}{\partial A} = 0. \]

During the calculation, it should be noticed that
\[ \int f(x_i | s_t = m; \Lambda') dx_i = 1 \]
\[ \int x_i f(x_i | s_t = m; \Lambda') dx_i = \mu_m \]
\[ \int (x_i - \mu_m)^T f(x_i | s_t = m; \Lambda') dx_i = \Sigma_m \]

(15)
\[ (x_i - \mu_m)^T f(x_i | s_t = m; \Lambda') dx_i = \Sigma_m \]

Firstly, substituting equation (13) and (14) into equation (15), and then get
\[ \sum_{i,m} [\Delta \gamma(t,m)] A^{-1} [\Sigma_m^{-1/2} (x_i - \mu_m) - b] + \sum_{i,m} d'(t,m) A^{-1} b = 0 \]

(16)

From equation (20), we can work out the bias of the mean parameters of the HMMs as follows,
\[ b = \sum_{i,m} \Delta \gamma(t,m) \Sigma_m^{-1/2} (x_i - \mu_m) \]
\[ \sum_{i,m} [\Delta \gamma(t,m) + D(m)] \]

where \( D(m) = \sum_{i,m} d'(t,m) \), in the next section, we will give its calculation equation.

As for the calculation of equation (16), it will be complexity but with the same method. It is easy to get the close form solution of A as follows,
3.2. Adaptation experiments and Results

Starting from the baseline system, we conduct supervised batch-mode adaptation experiments on four group testing data by using MLLR algorithm [1], MCELR algorithm with only mean adaptation [3] and the aforementioned IMCELR algorithm respectively. The iteration number is set to four. The test results are illustrated in the table 1 to table 4, and table 1 and table 2 evaluated the recognition performance of the proposed algorithm when the noise type is white noise and factory noise, and the SNR is 0dB, and compared with MLLR mean only adaptation and MCELR mean only adaptation. And table 3 and table 4 evaluated the recognition performance with the same noise type but the SNR is 6dB. As we expected, the proposed algorithm can further reduce the word error ratio of the recognition system.

Table 1: Recognition performance (WER%) when noise type is white and the SNR is 0dB

<table>
<thead>
<tr>
<th>Num Sample</th>
<th>0</th>
<th>10</th>
<th>50</th>
<th>100</th>
<th>300</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLLR</td>
<td>60.33</td>
<td>52.93</td>
<td>55.07</td>
<td>56.08</td>
<td></td>
</tr>
<tr>
<td>MCELR</td>
<td>58.60</td>
<td>51.33</td>
<td>50.76</td>
<td>49.67</td>
<td></td>
</tr>
<tr>
<td>IMCELR</td>
<td>81.73</td>
<td>57.46</td>
<td>49.78</td>
<td>46.20</td>
<td>46.60</td>
</tr>
</tbody>
</table>

Table 2: Recognition performance (WER%) when noise type is factory and the SNR is 0dB

<table>
<thead>
<tr>
<th>Num Sample</th>
<th>0</th>
<th>10</th>
<th>50</th>
<th>100</th>
<th>300</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLLR</td>
<td>57.73</td>
<td>56.20</td>
<td>55.27</td>
<td>54.53</td>
<td></td>
</tr>
<tr>
<td>MCELR</td>
<td>54.67</td>
<td>49.60</td>
<td>49.30</td>
<td>47.73</td>
<td></td>
</tr>
<tr>
<td>IMCELR</td>
<td>65.67</td>
<td>53.23</td>
<td>48.76</td>
<td>46.83</td>
<td>46.90</td>
</tr>
</tbody>
</table>

Table 3: Recognition performance (WER%) when noise type is white and the SNR is 6dB

<table>
<thead>
<tr>
<th>Num Sample</th>
<th>0</th>
<th>10</th>
<th>50</th>
<th>100</th>
<th>300</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLLR</td>
<td>65.00</td>
<td>55.20</td>
<td>39.20</td>
<td>40.93</td>
<td></td>
</tr>
<tr>
<td>MCELR</td>
<td>56.20</td>
<td>54.60</td>
<td>41.70</td>
<td>40.33</td>
<td></td>
</tr>
<tr>
<td>IMCELR</td>
<td>61.87</td>
<td>57.60</td>
<td>51.40</td>
<td>36.10</td>
<td>35.60</td>
</tr>
</tbody>
</table>

Table 4: Recognition performance (WER%) when noise type is factory and the SNR is 6dB

<table>
<thead>
<tr>
<th>Num Sample</th>
<th>0</th>
<th>10</th>
<th>50</th>
<th>100</th>
<th>300</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLLR</td>
<td>57.00</td>
<td>51.53</td>
<td>48.13</td>
<td>37.27</td>
<td></td>
</tr>
<tr>
<td>MCELR</td>
<td>59.76</td>
<td>52.30</td>
<td>46.56</td>
<td>42.61</td>
<td></td>
</tr>
<tr>
<td>IMCELR</td>
<td>58.33</td>
<td>49.70</td>
<td>45.30</td>
<td>40.93</td>
<td></td>
</tr>
</tbody>
</table>

4. Summary

We have presented a new formulation of MCELR for CDHMM adaptation both for mean and covariance parameters simultaneously. The linear transformation parameters are deliberatively selected and after some mathematical processing, we have derived the close form solution. The proposed algorithm provides an effective solution to apply MCE criterion to acoustic model adaptation with sparse data. By a series experiments, all tasks show that the proposed algorithm, IMCELR, is efficiency on the model adaptation under different environments and achieved significant performance advantage over the MLLR and MCELR with only mean adaptation approaches.

5. Reference