Discriminative pronunciation modeling based on minimum phone error training

Meixu Song, Qingqing Zhang, Jielin Pan, Yonghong Yan

Key Laboratory of Speech Acoustics and Content Understanding
Chinese Academy of Sciences
Beijing 100190, China
{songmeixu, zhangqingqing, panjielin, yanyonghong}@hccl.ioa.ac.cn

Abstract

Introducing pronunciation models into decoding has proven beneficial for LVCSR. As Minimum Phone Error (MPE) training has almost become a standard scheme for acoustic modeling, a discriminative pronunciation modeling method is investigated under the framework of MPE training. In order to bring the pronunciation models into MPE training, the auxiliary function of MPE training is rewritten at word level, and decomposes into two parts. One is for co-training the acoustic models, and the other is for discriminatingly training the pronunciation models. On Mandarin conversational telephone speech recognition task, compared to the baseline using a canonical lexicon, the discriminative pronunciation models reduced the absolute Character Error Rate (CER) by 0.7% on LDC test set, and with the acoustic model co-training, about 1% additional CER decrease had been achieved.

Index Terms: pronunciation models, discriminative training, Mandarin conversational speech recognition

1. Introduction

Current LVCSR technology aims at transferring real-world speech to sentence. Due to data sparsity, it is almost impossible to find a well direct transition between speech and sentence. Therefore, this transition is divided into three parts, as show in Figure 1: (a) the transition between speech feature vectors and subwords (phones for example) described by Acoustic Models (AMs); (b) the transition between words and sentence described by Language Model (LM); and (c) the transition between subwords and words described by a lexicon. We consider that a lexicon is composed of three parts: words, pronunciations, and Pronunciation Models (PMs). PMs contain the pronunciation probabilities of each word in the lexicon. In many LVCSR systems, the lexicon is hand-crafted, that usually means the pronunciations are in canonical forms, and the probability in PMs could be considered as constant 1. To automatically learn a lexicon, earlier studies have explored the data-driven pronunciation learning and PM training methods.

As to pronunciation learning, the work of [1] presents a discriminative pronunciation learning method using phonetic decoder and minimum classification error criterion. And the work of [2][3] makes use of a state-of-the-art letter-to-sound (L2S) system based on joint-sequence modeling [4] to generate pronunciations. Specifically for Mandarin pronunciation learning, the pronunciation variants of each constituent character in a word are enumerated to construct a pronunciation dictionary in [5]. This method is used to generate pronunciations for words in this paper, and the implementation details will be described.

\[ \hat{W} = \arg \max_{W} P(O|W)P(W) \]

As to PM training, in [2][4], a pronunciation mixture model (PMM) is presented by treating pronunciations of a particular word as components in a mixture, and the distribution probabilities are learned by maximizing the likelihood of acoustic training data. By contrast, in our work we modify the auxiliary function of standard MPE training [6] to incorporate PMs. By doing so, a discriminative pronunciation modeling method using minimum phone error criterion is proposed, called as MPEPM.

2. Pronunciation Models

With PMs considered in speech recognition, the most likely words sequence using Viterbi approximation is [1, 7]:

\[ P(B|W) = P(b_1, \ldots, b_k, w_1^t, \ldots, w_k^t) = \prod_{j=1}^{k} P(b_j|w_j^t) \]
\( P(b_j|w_j') \) is the probability that the \( j \)-th word in the \( r \)-th word sequence with \( k_r \) words, is pronounced as \( b_j \).

### 3. Incorporate PMs into MPE training

To train PMs from speech corpus, the possible pronunciations for each training utterance are usually required. In [3], these are obtained by decoding the N-best list of pronunciations. By contrast, the possible pronunciations sequences are already contained in lattices used in standard MPE training. Thus, the incorporation of PMs into MPE training is investigated.

The MPE objective function is [6]:

\[
\mathcal{F}_{\text{MPE}} = \sum_{r=1}^{R} \sum_{w=1}^{W_r} P_b(O_r, P_b(W_r) P(W_r) A(W_r))
\]

where \( A(W) \) represents the phone accuracy calculation function. To make (2) tractable, the auxiliary function of the MPE objective function is:

\[
\mathcal{H}_{\text{MPE}}(\lambda, \lambda') = \sum_{r=1}^{R} \sum_{w=1}^{W_r} \gamma_{\lambda, \lambda'}(\mathcal{F}_{\text{MPE}}) \log P(q)
\]

where \( P(q) \) is the likelihood of the speech data aligned to phone arc \( q \), \( \gamma_q \) is the posterior probability of the phone arcs \( q \) in current lattice, \( c_q \) (\( q \)) is the average phone accuracy of paths passing through the phone arcs \( q \), and \( c_{\text{avg}} \) is the average phone accuracy of all paths in the lattice of the \( r \)-th utterance [6].

To incorporate PMs, \( P(O_r|B) P(B|W) \) as in (1), then the MPE objective function is:

\[
\mathcal{F}_{\text{MPE}} = \sum_{r=1}^{R} \sum_{w=1}^{W_r} P_b(O_r, B_r) P(B_r|W_r) P(W_r) A(W_r)
\]

We rewrite auxiliary function (3) at word level and incorporate pronunciation probability \( P(b|w) \) as:

\[
\mathcal{H}_{\text{MPE}}(\lambda, \lambda') = \sum_{r=1}^{R} \sum_{w=1}^{W_r} c_{\lambda, \lambda'}(w, b_r) \log P(b(w)) P(b(w)|w)
\]

where \( (w, b) \) represents word \( w \) pronounced as \( b \). \( P(b) \) is the likelihood of the speech data aligned to word arc \( (w, b) \). \( W^R \) is the words set in the lattice of the \( r \)-th utterance. Accordingly, \( r_{(w, b)} \) is the posterior probability of the word arc \( (w, b) \) in current lattice. \( c_q \) (\( w \)) is the average phone accuracy of paths passing through the word arc \( (w, b) \), and \( c_{\text{avg}} \) is the average phone accuracy of all paths in the lattice of the \( r \)-th utterance [5]. (5) is based on a sum over word arcs \( w = 1 \ldots W^R \), each with start and end times.

By expanding \( \log P(b(w)) P(b(w)|w) \) to \( \log P(b(w)) + \log P(b(w)) \), (5) decomposes into two parts: AM co-training and PM training. The analyses of these two parts are as follows.

### 3.1. Co-train AMs

Suppose the pronunciation \( b \) for word \( w \) consists of phones \( q_{i1} \ldots q_{i\text{nw}} \), then

\[
\log P(b) = \sum_{i=1}^{\text{nw}} \log P(q_{i})
\]

where \( P(q_{i}) \) is the likelihood of the data aligned to phone arc \( q_{i} \). If the duration of \( q_{i} \) and \( q \) in (3) is equal, then \( P(q_{i}) = P(q) \).

Thus, (5) becomes:

\[
\mathcal{H}_{\text{MPE}}(\lambda, \lambda') = \sum_{r=1}^{R} \sum_{w=1}^{W_r} \gamma_{(w, b)} P_b(O_r, B_r) P(B_r|W_r) P(W_r) A(W_r)
\]

As the paths passing through word arc \( (w, b) \) are equal to those passing through any phone arc in word arc \( (w, b) \), namely for any \( q_{i} \in (w, b) \):

\[
\gamma_{(w, b)} = \gamma_{q_{i}} c_q (q_{i}) - c_{\text{avg}}
\]

Then the first part of (6) is:

\[
\sum_{r=1}^{R} \sum_{w=1}^{W_r} \gamma_{(w, b)} \sum_{i=1}^{\text{nw}} P_b(O_r, B_r) P(B_r|W_r) P(W_r) A(W_r)
\]

To keep statistics calculation consistent with that in standard MPE training, we will demonstrate the above formula (Equation (8)) is equal to the original auxiliary function (Equation (3)) with \( \gamma_{\text{MPE}} \) calculated as:

\[
\gamma_q = \frac{\partial \mathcal{F}_{\text{MPE}}}{\partial \log P(q)} = \sum_{w, q_{i} \in W_r} P_b(O_r, B_r) P(B_r|W_r) P(W_r) A(W_r)
\]

The first part of (9) is:

\[
\sum_{w_1, q_{i} \in W_r} P_b(O_r, B_r) P(B_r|W_r) P(W_r) A(W_r)
\]

By expanding \( \log P(q_{i}) \cdot \gamma_{q_{i}} \), each with start and end times.

The analyses of these two parts are as follows.
The second part of (9) equals to:
\[
\sum_{q^w} \sum_{W^i} P_x(O^i | B_i) P(B_i | W_i) P(W_i)
\]
\[
= \sum_{W^i, q^w} P_x(O^i | B_i) P(B_i | W_i) P(W_i)
\]
\[
= \sum_{W^i} P_x(O^i | B_i) P(B_i | W_i) P(W_i)
\]
\[
= \sum_{q^w} \sum_{W^i} \gamma_{q^w} \cdot c_{wq} \cdot P^{(q)}(\lambda_{wq} | o_{wq}) P(B_i | W_i) P(W_i)
\]
\[
= \sum_{q^w} \sum_{W^i} \gamma_{q^w} \cdot c_{wq} \cdot P^{(q)}(\lambda_{wq} | o_{wq})
\]
\[
= \sum_{q^w} \sum_{W^i} \gamma_{q^w} \cdot c_{wq} \cdot P^{(q)}(\lambda_{wq} | o_{wq})
\]
\[
(11)
\]
From (9)(10)(11) we have:
\[
\gamma^{MPE}_{\{w\}} = \sum_{q^w} \gamma^w_{q^w} \cdot (c^w_{q^w} - c_{wq})
\]
Finally, from (3)(12), we know (3) is a sum of \(\gamma^{MPE}_{\{w\}} \cdot (c^w_{q^w} - c_{wq})\) log \(P(q^w)\) over phone arcs, while (8) is a sum of the same thing over word arcs. The results are equal, namely:
\[
\sum_{r=1}^{R} \sum_{w=1}^{W^i} \gamma^w_{\{w\}} \sum_{i=1}^{Q^w} P(q^w)
\]
\[
= \sum_{r=1}^{R} \sum_{q^w=1}^{Q^w} \gamma^w_{\{w\}} \log P(q)
\]
\[
= \sum_{r=1}^{R} \sum_{q^w=1}^{Q^w} \sum_{i=1}^{Q^w} \gamma^w_{\{w\}} \cdot (c^w_{q^w} - c_{wq}) \log P(q)
\]
Therefore, using \(\gamma^{MPE}_{\{w\}}\) calculated by (9), AMs are co-
tained with PMs without changing MPE framework.

3.2. MPEPM

From (6)(7), we get the objective function of PMs using minimum 
phone error criterion:
\[
\sum_{r=1}^{R} \sum_{w=1}^{W^i} \gamma^w_{\{w\}} \log P^{(q)}(b | w)
\]
with constraints:
\[
P^{(q)}(b | w) = 1
\]
\[
P^{(q)}(b | w) \in [0, 1]
\]
We define
\[
\gamma^{num}_{\{w,b\}} = \gamma^w_{\{w\}} \text{ if } \gamma^{MPE}_{\{w,b\}} \geq 0
\]
\[
\gamma^{den}_{\{w,b\}} = \gamma^w_{\{w\}} \text{ if } \gamma^{MPE}_{\{w,b\}} < 0
\]
Referring to the auxiliary function used to update weight in 
MPE training [6], we use an auxiliary function for (13), that is:
\[
\sum_{r=1}^{R} \sum_{w=1}^{W^i} \gamma^w_{\{w\}} \log P^{(q)}(b | w)
\]
By maximizing this auxiliary function, the objective function 
in (13) is optimised with constraints of (14)(15). The detailed 
proofs could be found in [6]. For all \(\{w, b\}\), set \(P^{(0)}(b | w) =
\]
\[
P^{(q)}(b | w)
\]
, where \(P^{(q)}(b | w)\) is the probability in the former PMs.
And the iterative formula is as follows, in the \((p+1)\)-th iteration:
\[
P^{(p+1)}(b | w) = \frac{\gamma^{num}_{\{w,b\}} + k_b P^{(p)}(b | w)}{\sum_{b} \gamma^{num}_{\{w,b\}} + k_b P^{(p)}(b | w)}
\]
with
\[
k_b = \left( \frac{\gamma^{den}_{\{w,b\}}}{\sum_{b} \gamma^{den}_{\{w,b\}}} \right) - \frac{\gamma^{den}_{\{w,b\}}}{P^{(p)}(b | w)}
\]
The values of \(P^{(p+1)}(b | w)\) after 100 iterations are used as the 
updated pronunciation probabilities.

The above two subsections have shown the incorporation of 
PMs into MPE training. Through this, a discriminative pronun-
ciation modeling method is proposed. To compare this method 
with PMM [2, 3], a pronunciation model training method based 
on maximum likelihood criterion, we implement PMM training 
in MPE training at first in the next section, then we derive a 
relation between PMM and the proposed MPEPM.

4. PMM and MPEPM

In PMM , PMs is trained by maximizing the log-likelihood of 
the acoustic training data [2, 3]:
\[
P^{*}(b | w) = \arg \max_{P^{(b | w)}} \sum_{r=1}^{R} \log P(O_{r} | B_{r}, W_{r})
\]
\[
= \arg \max_{P^{(b | w)}} \sum_{r=1}^{R} \log \sum_{b} P(O_{r} | B_{r}) \prod_{j=1}^{k_r} P(b_j | w_j')
\]
\(P^{*}(b | w)\) could be optimized by using EM algorithm:
E-step:
\[
\mathbb{M}[b, w] = \sum_{r=1}^{R} \sum_{w} P(B_r | O_r, W_r) \mathbb{M}[b, w, W_r, B_r]
\]
M-step:
\[
P^{*}(b | w) = \frac{\mathbb{M}[b, w]}{\sum_{b, w} \mathbb{M}[b, w]}
\]
where \(\mathbb{M}[b, w, W_r, B_r]\) is the number of times word \(w)\)
appears in \(W_r\) aligned with the pronunciation \(b\). In [3], the 
possible pronunciation sequence \(B_r\) for words sequence \(W_r\) is 
contained in the N-best list of pronunciation sequences.

The update equation of PMM (17) actually is
\[
P^{*}(b | w) = \frac{\sum_{r=1}^{R} \text{the posterior of (w,b) in N-best list}}{\sum_{r=1}^{R} \text{the posterior of (w,b) in N-best list}}
\]
Naturally, the reference (correct transcription) lattices used in 
MPE training contain the possible pronunciation sequences. 
Thus we use lattices to replace the N-best list, the above equation 
becomes
\[
P^{*}(b | w) = \frac{\sum_{r=1}^{R} \text{the posterior of (w,b) in lattice}}{\sum_{r=1}^{R} \gamma_{\{w,b\}} \text{in lattice}}
\]
\[
= \frac{\sum_{r=1}^{R} \gamma_{\{w,b\}}}{\sum_{r=1}^{R} \gamma_{\{w,b\}}}
\]
\[
= \frac{\sum_{r=1}^{R} \gamma_{\{w,b\}}}{\sum_{r=1}^{R} \gamma_{\{w,b\}}}
\]
\[
(18)
\]

1943
where “ML” means the statistic is calculated on the reference lattices, and as in (7), we know γ\(^{ML}\)(w,b) = ∀qw\(_i\) ∈ {w,b}\(^{ML}\). Therefore we could train PMM by using statistics in MPE training.

To show a relation between PMM and MPEPM, we use the Extended Baum-Welch method [8] instead of the auxiliary function (16), to derive an update equation of P(b|w) from (13), that is:

\[
P^*(b|w) = \frac{\sum_{t} R(w,b) \cdot CP^*(b|w)}{\sum_{t} R(w,b) \cdot CP^*(b|w)}
\]

\[
= \frac{\sum_{t} R(w,b) \cdot CP^*(q_t^w) + C P^*(b|w)}{\sum_{t} R(w,b) \cdot CP^*(q_t^w) + C P^*(b|w)}
\]

where C is set to the smallest value necessary to make all updated P(b|w) positive. This update equation of P(b|w) is similar to that of weights in Maximum Mutual Information (MMI) training, thus the derivation details could be found in [6].

Comparing (18) and (19), except CP\(^{′}\)(b|w) in (19) which makes the constraint of (15) satisfied, we see two differences between PMM and MPEPM. The first difference is the lattices used to train PMs. While PMM calculates on the reference lattices, MPEPM calculates on reference and hypothesis (recognition) lattices. The second difference is the MPE discriminative factor “c\(^{′}\)(q\(_t^w\)) − c\(^{avg}\)” in MPEPM.

## 5. Experiments and Results

### 5.1. Construct Pronunciation Dictionary

We utilized the method employed in [5] to construct a pronunciation dictionary for 43k Chinese words. A character pronunciation dictionary with 7.8k pronunciations for 6.7k Chinese characters was used, to construct a full pronunciations set with 85k pronunciations. After performing a forced alignment of the acoustic training data, a 0.5 threshold relative to the maximum frequency of pronunciations of every word was set to prune out low frequent pronunciations. Finally, the pronunciation dictionary used to train PMs consisted of 47k pronunciations. The frequencies of remaining pronunciations of every word are normalized to form the initial pronunciation models.

### 5.2. Baseline System

Experiments were carried out on Mandarin conversational speech recognition task. The acoustic training data is about 400 hours, consisted of two parts. One is from LDC database including CallHome&CallFriend (45.9 hours), and LDC04 (150 hours) training sets. LDC04 was collected by Hong Kong University of Science and Technology (HKUST) in 2004. The other part is 200 hours speech data collected by all. The data were recorded through the landline telephone with local service in the real world with environmental noise. All utterances are in Chinese Mandarin and in spontaneous style.

The baseline acoustic models were trained by Minimum Phone Error (MPE) training [6]. All the triphone HMMs were 3-state left-to-right topology and used a two-level phonetic decision trees based state clustering [9]. Final acoustic models contained 7995 shared states with 16 Gaussians per state.

### 5.3. Recognition Results

There are two test sets. The first is “HTest04" collected by HKUST and released in 2005, which comprises of 4 hours of data with 24 phone calls. The second is “GDTest", comprised of half hour of self-collected data with 354 conversations by phone.

The recognition results are shown in Table 1. The first row is the result of baseline using a canonical lexicon. The second and third rows show the results of PMM and MPEPM without AM co-training, while the last row is the result of MPEPM with AM co-training. From these results, MPEPM shows its effectiveness, and with AM co-training, MPEPM outperforms PMM.

The MPEPM alone could not give an improvement over the PMM. We think the reason is the pronunciations per word (PPW) of MPEPM is 1.07, which is too small to show the superiority of discriminative method. Due to the pronunciation properties of Mandarin and simplicity of the adopted pronunciations generating method, we found no obvious improvement to augment the PPW directly. To adopt another sensible pronunciations generating method seems to be preferable.

### 6. Conclusions and Future Work

In this work, we presented a discriminative pronunciation modeling method based on MPE training. We rewrite the auxiliary function of MPE training at word level, and incorporated PMs into it. By doing this, we explored a way to discriminatively co-train the acoustic models and the pronunciation models in an iterative fashion. We demonstrated that the required statistics could be obtained in standard MPE training. Thus, this method is easy and efficient to implement. Furthermore, we showed relations and differences between this proposed MPEPM and PMM, both implemented in MPE training framework. In the end, experimental results on Mandarin conversational speech recognition task demonstrated the effectiveness of this method.

We hope to further our work in two directions. The first is to learn pronunciations for Chinese words from speech corpus, while joint-sequence models [4] used in [2, 3] do not support Chinese. The second is inspired by smoothing methods like H-criterion [10] and “I-Smoothing” [6][11] used in MPE training. We would like to explore an effective smoothing method for MPEPM.

### 7. Acknowledgements

This work is partially supported by the National Natural Science Foundation of China (Nos. 10905419, 90920302, 61072124, 11074275, 11161140319, 91120001, 61271426), the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant Nos. XDA06030100, XDA06030500), the National 863 Program (No. 2012AA012503) and the CAS Priority Deployment Project (No. KGZD-EW-103-2).

<table>
<thead>
<tr>
<th>Table 1: Results in CER (%)</th>
<th>HTest04</th>
<th>GDTest</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>49.7</td>
<td>50.8</td>
</tr>
<tr>
<td>PMM</td>
<td>49.1</td>
<td>50.3</td>
</tr>
<tr>
<td>MPEPM</td>
<td>49.0</td>
<td>50.5</td>
</tr>
<tr>
<td>co-train AMs &amp; MPEPM</td>
<td>48.2</td>
<td>49.7</td>
</tr>
</tbody>
</table>
8. References


