Performance Improvement of Connected Digit Recognition Using Unsupervised Fast Speaker Adaptation

Young Kuk Kim, Hwa Jeon Song, Hyung Soon Kim

Dep. of Electronics Engineering,
Pusan National University, Busan, Korea
{ykukim, hwajeon, kimhs}@pusan.ac.kr

Abstract

In this paper, we investigate unsupervised fast speaker adaptation based on eigenvoice to improve the performance of Korean connected digit recognition over the telephone channel. In addition, utterance verification is introduced into speaker adaptation to examine whether input utterance is appropriate to adaptation or not. Performance evaluation showed that the proposed method yielded performance improvements. We obtained 18%-22% string error reduction by the N-best-based fast speaker adaptation method with utterance verification using Support Vector Machine.

1. Introduction

Speaker adaptation contributing to better representation of speaker characteristics in a speech model is very useful tool to improve speech recognition performance. Typical adaptation methods include Maximum A Posteriori (MAP) adaptation [1], Maximum Likelihood Linear Regression (MLLR)[2] and speaker clustering. Among them, eigenvoice method, one of speaker clustering methods, is known to be advantageous in speaker adaptation because the number of estimated parameters is small [3].

In this paper, to improve the performance of Korean connected digit recognition over the telephone channel, we employ two approaches using eigenvoice-based instantaneous speaker adaptation method [5], in which each recognition utterance by itself is used to estimate the adaptation transform. In addition, utterance verification is introduced into speaker adaptation to examine whether input utterance is appropriate to adaptation or not. The first approach rescores alternative hypothesis from N-best decoding results with eigenvoice adapted model for input utterance with low confidence [6]. In second method, high confidence utterance from utterance verification is used to perform adaptation. In the case of utterance with low confidence, only matched frames between 1st and 2nd decoding hypotheses are used to perform adaptation.

In next section, we describe our baseline Korean connected digit system for evaluating our methods. Section 3 provides a description of unsupervised fast speaker adaptation based on eigenvoice using utterance verification. Section 4 shows experimental results, and conclusion is drawn in section 5.

2. Korean connected digit recognition system

2.1. Baseline system

Our baseline system used triphone with continuous mixture density HMM. Each HMM has three to five states and the number of mixtures per state varied from 1 to 20. We tied the states using tree based clustering (TBC) method.

Since all Korean digits are composed of one syllable and many of them are acoustically very confusable, Korean connected digit recognition is very difficult task. We chose optimal phone set to deal with co-articulation effects in Korean digit strings [7].

2.2. Baseline experiment and results

Korean connected digit database were collected by Electronics and Telecommunications Research Institute (ETRI) over the telephone line and sampled at 8 kHz [8]. Among them, we used three, four, six, seven, and ten-digit strings for training and test. We trained SI model using 47,858 utterances and used remaining 11,796 utterances for evaluation. The speech data was segmented into 20ms frame at every 10ms. We used 38 dimensional feature parameters (12 MFCCs, deltas, double deltas, delta energy, and double delta energy). Cepstral mean subtraction is performed for each utterance.

We performed baseline experiments according to different phone set, mixture number, and state number and the best results for 3 states and 5 states per phone cases are shown in Table 1.

<table>
<thead>
<tr>
<th>Experimental Condition</th>
<th>String Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 digit</td>
<td>88.99</td>
</tr>
<tr>
<td>10 digit</td>
<td>73.14</td>
</tr>
<tr>
<td>3 states (15 phone set, 17 mixtures)</td>
<td>92.29</td>
</tr>
<tr>
<td>5 states (14 phone set, 17 mixtures)</td>
<td>77.93</td>
</tr>
</tbody>
</table>

According to our experiments, 5 states per phone model and 14 phone set with 17 mixtures per tied state showed the best recognition performance. We used this optimized condition set in our unsupervised fast speaker adaptation experiments.
3. Eigenvoice adaptation

3.1. Unsupervised fast speaker adaptation system

Unsupervised fast speaker adaptation system is shown in Fig. 1. First, multiple word sequence hypotheses of the input speech are obtained by N-best decoding using SI HMM models. And then we decide whether the 1st recognition result is correct or not using utterance verification. If the result has low confidence score, we perform 2nd recognition after fast speaker adaptation. On the other hand, if the result has high confidence, then we accept the result without adaptation. In section 3.5, we will describe the method of utterance verification we used.

3.2. Eigenvoice adaptation

Eigenvoices are basis vectors in eigen space effectively representing the distribution of deviation of a large number of training speakers. Eigenvoice adaptation represents a new speaker with weighted sum of K eigenvoices as

\[ \hat{\mu} = e(0) + \sum_{k=1}^{K} w(k)e(k) \]  

where e(0) is the mean of T speaker dependent (SD) models, and w(k) is the weight of e(k), the k-th eigenvoice. The weights can be estimated by Maximum Likelihood Eigen-Decomposition (MLED)[3][4] using adaptation data of a new speaker.

3.3. Method 1: N-best-based fast speaker adaptation

N-best decoding based fast speaker adaptation is shown in Fig. 2. This approach assumes that there exists correct string within the N hypothesis sequences. Therefore if we rescore alternative hypothesis from N-best decoding result with eigenvoice adapted models for input utterance with low confidence, the correct string may have best likelihood score.

3.4. Method 2: Fast speaker adaptation using confidence region

Fast speaker adaptation using confidence region is shown in Fig. 3. In method 2, if input utterance has high confidence, we use the whole utterance to perform adaptation. In the case of utterance with low confidence, only matched frames between 1\textsuperscript{st} and 2\textsuperscript{nd} decoding hypotheses is used to perform adaptation.

![Figure 1: Unsupervised fast speaker adaptation with UV](image)

![Figure 2: N-best decoding based fast speaker adaptation](image)

![Figure 3: Fast speaker adaptation using confidence region](image)
Method 2 comprises of four steps:
1. First, multiple word sequence hypotheses of the input speech \( \{S_1, S_2, ..., S_N\} \) are obtained by N-best decoding using SI HMM model.
2. Examine whether input utterance has high confidence or not using the following equation.
   \[
   \frac{1}{T_1} \log P(O \mid \Lambda_{1_{-best}}) - \frac{1}{T_2} \log P(O \mid \Lambda_{2_{-best}}) > TH
   \] (2)
   where \( T_1 \) and \( T_2 \) are frame duration of 1st and 2nd best candidates, and \( P(O \mid \Lambda_{1_{-best}}) \) and \( P(O \mid \Lambda_{2_{-best}}) \) likelihoods of 1st and 2nd best.
3. If input utterance has high confidence, whole utterance is used to perform adaptation. Otherwise only matched frames between 1st and 2nd hypotheses are used for adaptation.
4. Finally, we perform 2nd recognition using SA models.

3.5. Utterance Verification

3.5.1. Confidence measures

The features we use for confidence measurement in our study are based on anti-digit models and N-best decoding information.

In the case of using anti-digit models [9], we used equation (3) as confidence measure for J-digit string.

\[
S(O; A) = -\log \left[ \frac{1}{J} \sum_{q=2}^{J} \exp \left[ -\eta \cdot LR_q(O;A) \right] \right]^{1/J} \] (3)

where

\[
LR_q(O; A) = g_q(O; A) - \left[ \frac{1}{N-1} \sum_{j=\neq q} \exp \left[ \kappa \cdot g_j(O; A) \right] \right]^{1/k} \] (4)

and

\[
g_q(O; A) = \frac{1}{T_i} \log \left[ p(O \mid \Lambda_q) \right] \] (5)

\( J \) is the string length, \( \eta \) a positive constant, \( LR_q(O; A) \) digit-level log likelihood ratio, \( N \) the anti-digit model number, \( \kappa \) a positive constant, and \( g_q(O; A) \) is the likelihood score of digit \( q \).

In the case of using N-best decoding information, we used score difference, number of difference hypothesis, and overlap segment as confidence measures. Score difference between 1st best likelihood and mean of the rest likelihood scores from N-best decoding results is given by

\[
\text{Score difference} = \frac{1}{T_{i_{-best}}} \log p(O \mid \Lambda_{1_{-best}}) - \frac{1}{N-1} \sum_{j=\neq i}^{N} \frac{1}{T_{i_{-best}}} \log p(O \mid \Lambda_{j_{-best}}) \] (6)

where \( T_{i_{-best}} \) is frame duration of \( i_{-best} \) result, and \( N \) is the number of N-best hypotheses.

From the digit strings of N-best decoding sequences, we can find out how many alternative digits are hypothesized for each digit in the best hypothesis. If a majority of alternative digits whose segment overlaps with that of a digit in the best hypothesis have the same identity, it is reasonable to assume this digit is probably correctly recognized. Based on this idea, we propose two additional confidence measures: number of different hypotheses and overlap segment as follows:

\[
\text{Different hypotheses} = \sum_{i=2}^{N} \left( 1 - \delta(1 - \text{best}, i - \text{best}) \right) \times \frac{1}{i-1} \] (7)

\[
\text{Overlap segment} = \sum_{i=2}^{N} \delta(1 - \text{best}, i - \text{best}) \times \frac{d_i}{d_i} \] (8)

where

\[
\delta(1 - \text{best}, i - \text{best}) = \begin{cases} 1 & \text{if } 1 - \text{best} \text{ and } i - \text{best} \text{ are same hypothesis} \\ 0 & \text{if } 1 - \text{best} \text{ and } i - \text{best} \text{ are different hypothesis} \end{cases} \] (9)

\( N \) is the alternative hypotheses number and \( d_i \) is number of overlapping frames between the best and alternative hypotheses.

3.5.2. Classifiers

The features defined in 3.5.1 can be input to any statistical pattern classifier to decide whether to reject an utterance or not. In this paper, we focused our attention on the Neural Networks (NN) and Support Vector Machine (SVM) [10] as classifiers.

First, we used multi-layer perceptron neural networks to train the network performing classification. The structure of the neural network consists of 5 input neurons with bias and 2 output neurons. The conventional back-propagation (BP) algorithm was used to train the network and sigmoid function was used as activation function. The running rate, the moment coefficient, and hidden layer node were optimized by grid search.

Secondly, SVM is used as another classifier. An SVM learns the decision boundary between two classes by mapping the training sample vectors onto a higher dimensional space and then determining an optimal separating hyper-plane. In the case where a linear boundary is inappropriate, the SVM can map input vector into a high dimensional space through kernel function, where it can construct a linear hyper-plane in the high dimensional space. Since finding the SVM solutions involves the dot products of the sample vectors, kernel function is very important role. Radial basis function that has been shown to perform well in many classification tasks was selected as kernel functions. Radial basis function is of the form

\[
k(x, x_i) = \exp(-\|x - x_i\|^2 / 2\sigma^2) \] (10)
where $X_i$ denotes the support vector (SV), $X$ the feature vector to be classified, and $\sigma$ is the variance.

4. Experiments

4.1. Experimental condition

We used the same database for the baseline experiment in section 2. 4-digit and 10-digit strings are used (2,737 and 2,818 utterances) for fast speaker adaptation experiment. One thirds of DB is used for development set to train NN and SVM and remaining two thirds of DB is used as test set. A set of 829 SD model was constructed by MLLR and MAP adaptation with SI model. To obtain eigenvoice, we applied principal component analysis (PCA) to 829 SD models. And then we used 20 eigenvoices for adaptation. The dimension of supervector $T$ is 474,810 when we use 735 tied states, 17 mixtures, and 38 feature dimension.

4.2. Unsupervised speaker adaptation experiments

Table 2 shows the performance of unsupervised fast speaker adaptation. In this table, “Baseline” indicates the baseline performance with optimized condition in section 2.2, and “No UV” means that unsupervised fast speaker adaptation was performed for all test utterances without utterance verification. “No UV” yielded string error reduction of 18% and 16% for four-digit and ten-digit strings, respectively.

Table 2: Performance of unsupervised fast speaker adaptation

<table>
<thead>
<tr>
<th>Experimental Condition</th>
<th>SD Model</th>
<th>String Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>-</td>
<td>91.73 76.37</td>
</tr>
<tr>
<td>No UV</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Method1</td>
<td>NN + SVM</td>
<td>93.21 80.26</td>
</tr>
<tr>
<td>Method2</td>
<td>UV</td>
<td>92.82 78.02</td>
</tr>
</tbody>
</table>

“Method 1” and “Method 2” are N-best-based speaker adaptation method and speaker adaptation method using confidence region as described in section 3.5. Both methods outperformed the baseline system. “Method 2” showed higher performance than “Baseline” and “No UV”, but lower than “Method 1” because of simply using conventional LLR method as utterance verification. In “Method 1”, SVM classifier yielded higher performance than NN classifier. In the case of “Method 1” with SVM classifier, we obtained the best string error reduction of 22% and 18% in 4-digit and 10-digit strings respectively.

5. Conclusions

In this paper, we investigated two instantaneous speaker adaptation methods based on eigenvoice to improve the performance of Korean connected digit recognition over the telephone channel.

One is N-best-based fast speaker adaptation that rescores alternative hypothesis from N-best decoding results, and the other is fast speaker adaptation using confidence region in which only matched frames between 1st and 2nd decoding hypotheses are used. According to our experiments, the former showed higher performance than the later. In N-best based fast speaker adaptation with SVM classifier as utterance verification tool, we obtained 18%-20% string error reduction.

6. Acknowledgement

This work was supported by Speech/Language Information Research Department, the Electronics and Telecommunications Research Institute (ETRI), Korea.

7. References