Speech Recognition Using an Enhanced FVQ Based on a Codeword Dependent Distribution Normalization and Codeword Weighting by Fuzzy Objective Function

Hwan Jin Choi, Yung Hwan Oh
Department of Computer Science
Korea Advanced Institute of Science and Technology
373-1 Yusong-gu Kusong-dong
Taejon, Korea
e-mail : hjchoi@bulsai.kaist.ac.kr

Abstract

The paper presents a new variant of parameter estimation methods for discrete hidden Markov models(HMM) in speech recognition. This method makes use of a codeword dependent distribution normalization(CDDN) and a distance weighting by fuzzy contribution in dealing with the problems of robust state modeling in a FVQ based modeling. The proposed method is compared with the existing techniques using speaker-independent phonetically balanced isolated words recognition. The results have shown that the recognition rate of the proposed method is improved 4.5% over the conventional FVQ based method and the distance weighting to the smoothing of output probability is more efficient than the distance based codeword weighting.

1. INTRODUCTION

With the advent of new technologies, auditory models and the establishment of psychological cognitive models, the interesting of speech recognition is greatly increasing. Speech recognition requiring multi-level knowledge such as acoustics, phonetics, and linguistics is a process that extracts linguistic information from speech and converts it to understandable representations. Due to the variability, redundancies, massive computation, and a lack of any comprehensive theories in speech processing, speech recognition still remains a difficult problem[1]. For speech recognition, several approaches are being used and one of them, the discrete hidden Markov model, has been successfully applied to speech recognition. The discrete HMM[2] is attractive in its low cost of computation and high versatility, and therefore has been investigated in a number of studies. Nevertheless, one of the main disadvantages is the implicit discretization of the observations which produces information loss that in turn cause the model's performance to deteriorate. To alleviate these problems, the detailed state modeling with continuous density functions[3], smoothing of output probabilities[4] to remedy the sparseness of training data , and discriminative training[5][6] for recognition models.

The paper, we focuses on the reduction of quantization error and improving the state output probabilities in a state for a discrete hidden Markov model. To reduce the quantization error in DHMM, we have used a FVQ based state modeling. The FVQ based state modeling has shown improved recognition results [7][8]. But, there are some limitations in a state modeling with a FVQ. In a conventional FVQ based recognition, the distance measure is based on a relative Euclidean distance , and the unestimated output probabilities decrease the recognition rates. To alleviate all these problems, we have proposed a codeword based distance normalization to reflect the detailed characteristics of codeword and have used a smoothing method to prevent the degradation of the performance from unestimated output probabilities.

The paper is organized as follows. In section 2, we describe the definition of DHMM(discrete hidden Markov model) and the FVQ(fuzzy vector quantization), which are the base-form of the proposed method. In section 3, the codeword dependent distribution normalization and codeword weighting and smoothing by distance based fuzzy contribution are described. In section 4, we present the experimental environment used for testing the proposed recognition system. Finally in section 5, we summarize the conclusions and future works.

2. REVIEW OF FVQ BAED DHMM

2.1 Definition of DHMM

The hidden Markov model **S** is usually defined as a 5-tuples,(Q,V,π,A,B). *Q* is a set of *N* states $q_1,q_2, ...,q_N$. *V* is a set of

M symbols $v_1, v_2, ..., v_M$ representing a prototypical spectra. π is a vector which specifies the initial distribution ($\pi_1, \pi_2, ..., \pi_N$), where π_i = Prob($q_i(1)$). A is a matrix of state transition probabilities, $A = [a_{ij}]$, $1 \le i,j \le N$, where a_{ij} = Prob($q_j(t+1) \mid q_i(t)$). B is a matrix of observation probabilities, $B = [b_{ij}], 1 \le i \le N, 1 < j < M$, where b_{ij} = Prob($v_j(t) \mid q_i(t)$). The sequence of observations is denoted as $\boldsymbol{O} = (o_1, o_2, ..., o_T)$, where o_t for $1 \le t \le T$ is some $v_i \in V$. We are then interested in calculating Prob(**O**). This is usually done using the forward-backward or Viterbi algorithm. The computation of Prob(**O**) then follows as

$$\Pr ob(O) = \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) a_{ij} b_{jO_{t+1}} \beta_{t+1}(j) \qquad (1)$$

for any t such that $1 \le t \le T-1$.

2.2 FVQ(Fuzzy Vector Quantization) Based State Modeling

Fuzzy VQ can be viewed as a simplified case of the mixture Gaussian VQ. Let $d(x_i, c_j)$ represent the Euclidean distance between input vector x_i and codeword c_j. The FVQ maps an input vector x_i into an output vector $o_i = (m_{i1}, m_{i2}, ..., m_{iM})$ according to the rule :

$$m_{ij} = \left[\sum_{k=1}^{M} \left[d(x_i, v_j) / d(x_i, v_k) \right]^{1/(F-1)} \right]^{-1} \qquad (2)$$

where, $F > 1$ is a constant called the degree of fuzziness. Vector o_i is chosen in this way because it minimizes the fuzzy objective function

$$\sum_{i=1}^{T} \sum_{j=1}^{M} m_{ij}^{F} \cdot d(x_i, v_j) \qquad (3)$$

The components of o_i are positive and sum to 1.

When the observation is fuzzy, the observation sequence is a sequence of probability mass vectors. Denote this fuzzy observation sequence again by $\boldsymbol{O} = (o_1, o_2, ..., o_T)$, where each o_i is now a probability mass vector of the form $o_i = (m_{i1}, m_{i2}, ..., m_{iM})$. In this point, the estimation procedures must be modified to include this fuzzy observation. Let us define $\omega_t(i)$ for $1 \le t \le T$ and $1 \le i \le N$ to be Prob($o_t \mid q_i(t)$). Then, $\omega_t(i)$ is calculated using the equation

$$\omega_t(i) = \sum_{j=1}^{M} m_{ij} b_{ij} \qquad (4)$$

From this definition, we can compute the modified Viterbi algorithm. The forward probability at time t $\tilde{\alpha}_t(i)$ as follows.

$$\alpha_t(i) = \max_{1 \le i \le N} \left[\alpha_t(i) a_{ij} \right] \omega_t(j), \ 2 \le t \le T, \ 1 \le j \le N \qquad (5)$$

where, $\tilde{\alpha}_t(i) = \pi_i \omega_1(i)$ for all i. The final result is $\max_{1 \le i \le N} \tilde{\alpha}_T(i)$, as usual.

2.3 Problems in a FVQ Based Output Probability Modeling

As we have mentioned in the Introduction, the conventional FVQ based state output estimation method has some limitations. First, the characteristics of codeword dependent distributions are ignored. The distance in a state is a relative distance between an input vector and a codeword vector based on Euclidean distance. If we consider the distribution of input patterns in a codeword, the smallest distance is not always true. It can be the longest distance when we consider the characteristics of a distribution. Therefore, the detailed modeling of a distance in a state is required. Here, we have used the codeword dependent distribution and the distribution is defined as a normal distribution with a mean and a variance on distances for all of training data included in a state.

The second problem is that the distance between an input vector and a codeword is equally weighted. The importance of each codeword is different in a state. These differences are reflected in the output probability computation. Based on this assumption, we have derived the weighting factor of a codeword in a state with a distance based fuzzy contribution. this approach has a limitation. This proposed method does not resolve the unseen data problem. To alleviate this problem, the smoothing of a output distribution is required. The proposed method has a form which is a linear combination of multiplying a best codeword dependent fuzzy contribution distribution with a state dependent codeword probability. The detailed algorithms and their operations will be described in section 3.

3. CODEWORD DEPENDENT DISTRIBUTION NORMALIZATION AND DISTANCE WEIGHTING

In this section, we are going to describe the proposed approaches to alleviate some problems presented in the 2.3. The first subsection describes the codeword dependent distribution normalization for distance ,and the second subsection describes the codeword weighting and smoothing by a fuzzy objective function in modeling of state output probabilities.

3.1 Codeword Dependent Distribution

Normalization(CDDN)

To reflect the codeword dependent characteristics, we have computed the mean and the variance of distances between a codeword vector and input vectors in a training set(eqn. 6).

$$\mu_i = \frac{1}{P_i}\sum_{p}^{P_i} d(c(i), x_p(i))$$
$$\sigma_i^2 = \frac{1}{(P_i - 1)}\sum_{p}^{P_i} \frac{(d(c(i), x_p(i)) - \mu_i)^2}{{\mu_i}^2} \quad (6)$$

Where, c_i is a *i*-th codeword index, P_i is the number of training patterns corresponding to the *i*-th codeword vector. According to equation 6, the normalized probability for a distance $d(x_t,c(i))$ can be calculated as equation 7.

$$P(x_t, i) = \frac{(d(x_t, c(i)) - \mu_i)^2}{\sigma_i} \quad (7)$$

With the result of equation 7, we have transformed the codeword dependent probability for an input vector into the distance which reflects the characteristic of a corresponding codeword. Here, we assume the Gaussian model for a transformation.

$$D(x_t, i) = -1.0 \times \log_e(P(x_t, i)) \quad (8)$$

The final distance for a codeword in a state is equation 8.

3.2 Codeword Weighting and Smoothing by Fuzzy Contribution

To alleviate the problem of equal weighting, we have used fuzzy contribition based weighting factors. In a state, the best codeword distance for an input vector is accumulated and normalized. We denoted it as $\overline{D}(j)$, where *j* is the *j-th* codeword index. After that, we applied the fuzzy objective function to the normalized distance for each codeword. The derived fuzzy contribution for a *i-th* codeword is as following in equation 9.

$$W_i(s) = \left[\sum_{k=1}^{M}\left[\overline{D}(i) / \overline{D}(k)\right]^{1/(F-1)}\right]^{-1} \quad (9)$$

The distance of a codeword for an input vector ,which is calculated by equation 8, is finally weighted by a codeword dependent weighting value $W_i(s)$. The final distance value that will be used as input to equation 2 is taken by equation 10.

$$\tilde{D}(x_t, i) = D(x_t, i) \times W_i(s) \quad (10)$$

For the smoothing of output probabilities for unseen data, we define the matrix with N by N, where N is a size of a codebook. Given an input vector, the distance of each codeword is calculated and the best codeword is chosen from all of codewords. If the best codeword is chosen, we store the distance of others in a matrix given the best codeword. After all of training data in a state are received, we normalized the matrix by codeword and represented it as $\overline{D}$. To compute the fuzzy contribution for each codeword given a best codeword, we applied the fuzzy objective function to the matrix $\overline{D}$. Using the derived fuzzy contribution $m_{ij}(s)$ and the distribution of codewords in a state, we have calculated the smoothed output probability $m_i(s)$. This smoothed output probability replaces the original output probability in a state. The derived equation is in (11).

$$m_{ij}(s) = \left[\sum_{k=1}^{M}\left[\overline{D}(i,j) / \overline{D}(i,k)\right]^{1/(F-1)}\right]^{-1}$$
$$m_i(s) = \sum_{j=1}^{M}[P(j,s) \times m_{ji}(s)] \quad (11)$$

4. EXPERIMENTAL RESULTS

4.1 Experimental Environment

Comparative experiments have been carried out for evaluations. The speech data used in the experiments are the phonetically balanced 115 isolated words provided by ETRI(Electronic Telecommunication Research Institute), consisting of 3910 utterances of words. These utterances were spoken by 17 male speakers, each of whom provided 2 repetitions per words.

For speech recognition, the analog speech signals were converted to discrete-time data through an anti-aliasing low-pass analog filter and by using a 12 bit A/D converter with a 16 kHz sampling rate. The digitized speech signal was pre-emphasized with a first order filter whose transformation function was $1\text{-}0.97Z^{-1}$. The wave was then segmented into frames of 20msec(320 points) duration with 10 msec(160 points) overlap between consecutive frames. The Hamming window was used to remove the adverse effects of frame segmentation procedure. Each frame has been characterized by 14 order mel-cepstral coefficients(MCC),14-order delta mel-cepstral coefficients(DMCC), log energy and delta log energy. The LBG(Linda-Buzo-Gray) algorithm[9] was employed to generate three separate codebooks. The size of the codebook for MCC and DMCC vectors are 256. The feature parameter for energy and delta energy are merged and are taken as a single vector. The size of this codebook is 128. As a unit for recognition, we have used 50 context independent PLU(phone like unit). The PLU's are concatenated to a word model and each of PLU are trained with the segmental K-means algorithm[10]. Given a sequence of codewords, the recongizer selects a word

model with a maximum output probability from all of the word models.

To evaluate the performance of the proposed method, we have done the comparison experiments. The experiment results are shownin Figure 1. The CDDN method shows 95.7% and the conventional FVQ based state modeling shows a recognition rate of 94.3%. These experiments demonstrate the reflection of a characteristic of a codeword dependent distribution to the calculation of a output probability is more efficient than otherwise. The distance based codeword weighting methods also improved the recognition rate. Finally, we have applied the proposed smoothing method to the recognizer and received 98.8% recogntion rate. With these results, the smoothing of a distribution of output probability is more important than the codeword based weighting. In the experiments, we determined that the proposed methods are more efficient than the conventional approach in several of comparative results.

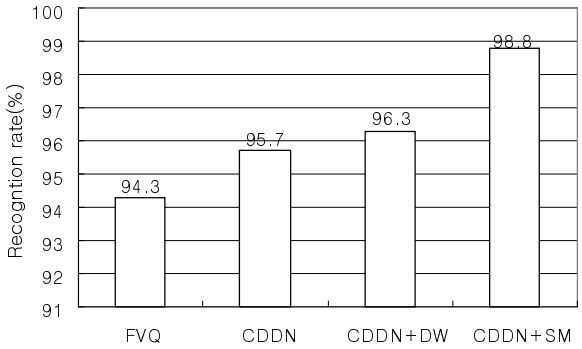

Figure 1: Experimental Results

5. CONCLUSION

A new parameter estimation method for discrete hidden Markov models has been presented. The proposed method has been evaluated on a task of speaker-independent isolated words recognition and was compared with the conventional approaches. From the experimental results, we have concluded as the following.

1. The information for codeword dependent distribution is effective in computing the output probability.

2. The codeword dependent weighing in a state has slightly improved performance.

3. Smoothing of output probability is more effective than the codeword dependent weighting in a state.

We have a plan to test the proposed approach with more training data and also will compare the proposed method with other kinds HMM models such as semi-continuous or continuous models for continuous speech recogntion.

REFERENCE

1. L. R. Rabiner, B. H. Juang, *Fundamentals of Speech Recognition*, Prentice-Hall International Editions, 1993.

2. Lee, K. F. , *Automatic Speech Recognition : The Development of the SPHNIX System*, Kluwer Publishers, Boston, 1989.

3. D. Huang, M. A. Jack,"Semi-Continuous Hidden Markov Models for Speech Signals", *Computer Speech and Language*, Vol. 3, No.2, pp. 239 - 251, 1989.

4. Schwartz, et al.,"Robust smoothing methods for discrete hidden Markov models", *Proc. of ICASSP*, Glasgow, Scotland, pp. 548 - 551, May 1989.

5. H. Applebaum, B. A. Hanson, "Enhancing the discrimination of speaker independent HMMs with corrective training", *Proc. of ICASSP*, pp. S6.13 - 14, 1989.

6. R. Bahl, et al.,"A new algorithm for the estimation of hidden Markov model parameters", *Proc. of ICASSP*, pp. S11.2 - 4, 1988.

7. M. Koo, C. K. Un, "Fuzzy smoothing of HMM parameters in speech recognition", *Electronic Letters*, Vol. 26, pp. 7443 - 744, 1990.

8. H. P. Tseng, M. J. Sabin, et al.,"Fuzzy Vector Quantization applied to hidden Markov modeling", *Proc. of ICASSP*, pp. 15.5.1 ~ 15.5.4, April 1987.

9. Gray, "Vector Quantization", *IEEE ASSP Magazine* , Vol. 1, pp. 4 - 28, April 1984.

10. L. R. Rabiner, J. G. Wilpon, B. H. Juang,"A Segmental K-means training procedure for hidden Markov models with continuous mixture densities", *AT & T Tech. Journal*, Vol. 65, pp. 21 - 31, 1986.