OPTIMIZATION OF PHONEME-BASED VQ CODEBOOK IN A DHMM SYSTEM

Yaxin Zhang, Roberto Togneri, Chris deSilva, Mike Alder
Center for Intelligent Information Processing Systems
Department of Electrical and Electronic Engineering
The University of Western Australia

Abstract

A phoneme-based Gaussian mixture VQ codebook can improve the conventional DHMM system performance significantly. In this paper, an optimization method for the phoneme-based VQ codebook is proposed. The experimental results shown that the optimized phoneme-based VQ codebook leads to both the improvement of system performance and the reduction of system complexity.

INTRODUCTION

In a phoneme-based Gaussian mixture codebook [1], each phoneme of each sex of speakers in the training sequence is represented by a codeword. This means that we treat the phonemes as data classes in the clustering by VQ. The number of codewords in a codebook, codebook size, is normally larger than the number of phonemes in the training sequence. The reason for this is that the phoneme space is only a subspace of the speech space. In other words, the speech signal space includes not only the phonemes, but also other sounds between the phonemes and silence or background noise. We generate a Gaussian mixture codebook in which some of codewords are determined by the individual phonemes and others by the rest of training sequence.

Investigations have shown that in phoneme-based Gaussian mixture codebook generated by the Expectation-Maximization (EM) algorithm [9], too many codewords are concentrated near the origin, the region of silence or background noise. For example, in a codebook of size 128, there are more than 10 Gaussians in this region. It is not reasonable to classify the silence into so many classes. Instead, the silence should be treated as one cluster. This not only reduces the system complexity, but also increases the cluster separability for the recognition.

In this paper we propose a phoneme-based VQ codebook optimization method to construct a minimal codebook to represent the complete speech data. In our experiments, a typical 5-state discrete HMM (DHMM) speech recognizer was employed for isolated word recognition of the letters of the English alphabet. The training and testing

method	DHMM			PBVQHMM		
codebook size	64	128	256	64	128	256
accuracies	78.9	83.5	86.4	92.7	95.2	94.1

Table 1: The accuracy rates of English alphabet recognition.

data were obtained from the TI46 database. The best results showed a 37.6% decrease in the codebook size while the recognition accuracy was increased by 1.32 percentage points.

PHONEME-BASED VQ FOR DHMM

The database used for training and testing was the TI46-Word Speaker-Independent Isolated Word Corpus from the National Institute of Standards and Technology (NIST) in the USA. The database comprises 46 isolated words, 10 digits, 10 computer command words, and 26 letters of the English alphabet. The data is sampled at 12500 Hz and digitized to 14 bit resolution. There are 16 speakers (8 male and 8 female) in the database and each word was repeated 26 times by the speakers. The first 10 repetitions were used as the training set and the remaining 16 as the testing set. Here we use the subset of the 26 letters only. The FFT of the speech data was computed every 10 ms and a 25 ms Hamming window was used. The FFT coefficients were binned into 12 Mel-spaced values to produce 12-dimensional feature vectors corresponding to the frequency range from 60 to 5000 Hz.

In the training procedure for the phoneme-based VQ codebook, the phonemes were manually extracted from the speech waveform of training sequence. There were 24 phonemes from this data for each sex. In the training procedure we estimated the parameters of two Gaussian models for each vowel or consonant, one for the data from the male speaker and one for the female speaker. For each diphthong, mixture model comprising two Gaussians were produced by the EM algorithm. This was based on the observation that the diphthong data look like two overlapped Gaussian clusters. From all the phonemes processed, we constructed 58 Gaussian models. The EM algorithm was employed to produce a Gaussian mixture model which included not only the Gaussian models for the individual phonemes, but also the Gaussian models which represented the sound between the phonemes and the background noise. The Gaussian mixture model was used in a traditional DHMM system as a VQ codebook. For comparison purposes, Gaussian mixture models with 64, 128, and 256 components were generated. In each of them, there were 58 components from the individual phonemes.

Table 1 is the comparison results of conventional discrete HMM (DHMM) recognizer and this phoneme-based VQ HMM (PBVQHMM) recognizer. It is obvious that the improvement of PBVQHMM system is significant.

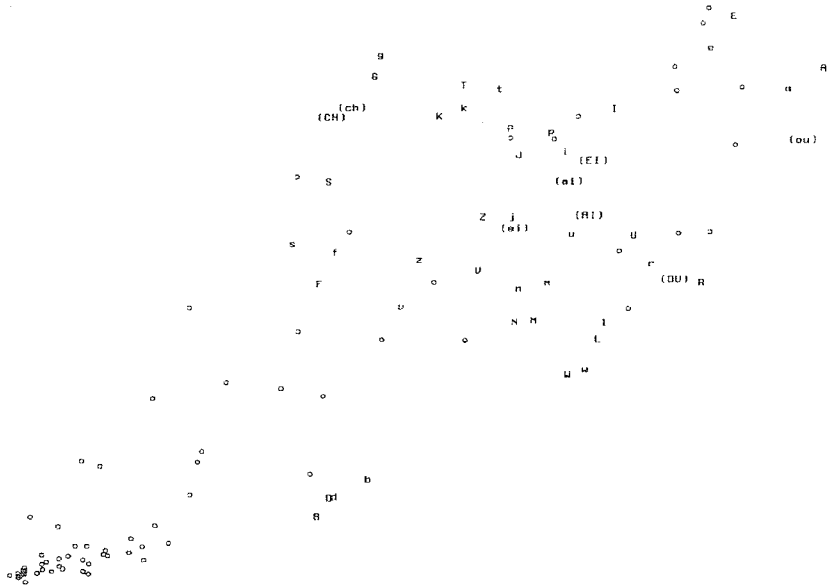

Figure 1: A projection of phoneme-based codebook

PROJECTION OF PHONEME-BASED VQ CODEBOOK

Figure 1 is a projection of the codebook on a two dimensional plane, in which the X-axis represents the average value of the first six components of the vector while the Y-axis represents the average value of the remaining components. The phoneme-based codebook was generated from the training sequence, and had 58 codewords from the phoneme modeling. Each symbol in Figure 1 indicates the position of the corresponding the phoneme. The capital letters indicate the positions for male speakers and the lowercase letters the positions for female speakers. To avoid confusion, we only use one letter pair to present the two code words from each diphthong. The circles represent the remaining codewords not derived from the phoneme modeling. These were generated by the EM algorithm from the whole training sequence. The picture is a representation of the speech space, at least from a two dimensional point of view. Considering the positions of the phonemes, we can say that the different classes of phonemes, vowels, stops, fricatives, and nasals, occupy separate regions in the space. It should be noted that most codewords not derived from the phonemes lie in the left-bottom area of the picture near the origin. These codewords represent the background noise or silence.

It appears that there are too many codewords concentrated in the region near the origin. It is not necessary to use so many Gaussian models to represent silence. However, when we reduce the codebook size in the EM training procedure, the recognition accuracies decrease. This is because there are reductions in both the codewords near the origin and also in the codewords in the phoneme area. The codewords in the phoneme area are very important for the recognition system since they represent the detailed sound between the phonemes. Table 2 indicates the changes of recognition accuracy

codebook size	80	90	100	110	128	140	200
accuracies	92.15	93.27	93.60	93.26	95.24	93.80	93.27

Table 2: The accuracy rates of English alphabet with different codebook size.

corresponding to changes in the codebook size. It shows that the best results are obtained when the codebook size is 128 and both increasing and decreasing the codebook size result in the system performance degradation.

OPTIMIZATION OF PHONEME-BASED VQ CODEBOOK

From Figure 1 we can see that more than 40 of the codewords in the phoneme-based codebook with 128 codewords are concentrated in the region near the origin. From a probability point of view, this is reasonable since the background noise contributes many more data points than any phoneme and has a greater variance than the speech data. This causes many Gaussians to be filted to the noise. From a speech recognition point of view, however, it is undesirable to have a large number of Gaussians representing silence and background noise. In fact we want a codebook in which each codeword efficiently represent a certain speech data class. Ideally, we need only one Gaussian to represent all the data associated with silence. It makes sense not only for reducing the codebook size thereafter reducing the system performance complexity, but also for reducing the confusions in the recognition process thereafter improving the recognition accuracy.

We propose a method for optimizing the phoneme-based VQ codebook. First we extract the silence data from the training speech data files. Then we estimate a single Gaussian model for the silence. To make sure no more Gaussians are generated in this region during the EM training procedure, we force all the silence data points to belong to the silence Gaussian. To avoid the data points associated with the sound between the phonemes being corporated into the silence Gaussian, this procedure must be based on the previous analysis which gives the definition of the range of silence data points. In this improved training procedure, we will produce an improved codebook in which each codeword efficiently represents a certain speech data class.

Figure 2 is a projection of the optimized phoneme-based VQ codebook. Comparing it to Figure 1, we can see that there is only one codeword in the left-bottom corner in this codebook while the codewords representing phonemes are unchanged.

Table 3 shows the results obtained from the experiments with optimized phoneme-based Gaussian mixture codebooks. The database and speech feature preparation are same as that described in Section 2. The first row indicates the codebook sizes before optimization and the second row shows the codebook sizes after optimization. For example, the codebook with 80 components was reduced to 74 components, the codebook with 90 components was reduced to 77 components, and so on. The third row shows the percentage reduction in codebook sizes. The fourth row gives the recognition accuracies of the recognizer with the optimized phoneme-based Gaussian mixture codebooks.

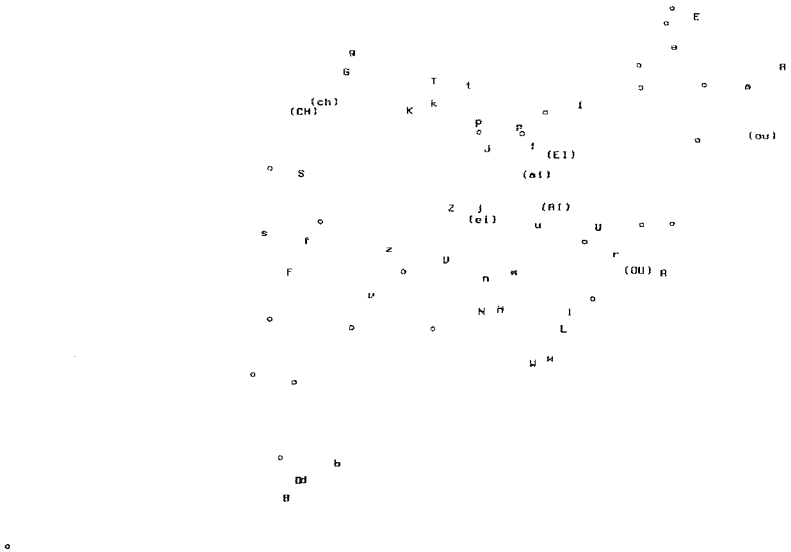

Figure 2: A projection of optimized phoneme-based codebook

original codebook size	80	90	100	110	128	140	200
optimized codebook size	74	77	78	80	81	90	131
codebook size reduction	7.5%	14.4%	22%	27.3%	36.7%	35.3%	34.5%
recognition accuracies	89.94	92.88	93.27	94.13	96.56	95.24	92.49
accuracy improvment	-2.21%	-0.39%	-0.33%	0.86%	1.32%	1.44%	-0.78%

Table 3: The performance results of the system with optimized phoneme-based codebooks

Finally, the improvements of system performance are shown in the fifth row, where negative value indicates a degradation of system performance.

CONCLUSIONS AND DISCUSSION

From these results we can make the following conclusions.

The PBDHMM speech recognition system with an optimal codebook can reduce the system performance complexity which includes both the computational cost and storage cost. For example, in the best result obtained in the case of original codebook size 128, both the computational cost of VQ process and storage cost are reduced by 36.7%.

The optimization of phoneme-based codebook is an efficient way to improve the system performance. In the best result obtained in the case of original codebook size 128, the recognition accuracy was increased by 1.32 percentage points. In other words, the recognition error rate is reduced by 28%. Compared with the conventional DHMM

system of which the results were shown in Table 1, we obtained a recognition accuracy gain of 13.1 percentage points or 80% recognition error rate reduction.

The optimal codebook size for isolated letter recognition should be in a range of 80 to 90. From Table 2 we can see that in this range the system performances are reasonably good and the accuracy improvments are positive.

ACKNOWLEDGEMENT

This work was supported in part by The University Fee-Waiver Scholarship and The University Research Studentship of the University of Western Australia.

REFERENCES

[1] Zhang, Y., Alder, M., & Togneri, R., "Using Gaussian Mixture Modeling in Speech Recognition", Proceedings of ICASSP 1994, April 1994, Adelaide, Australia. pp. I613-616.

[2] Huang, X. D. & Jack, M. A., (1988), "Hidden Markov Modeling of Speech Based on a Semicontinuous Model," Electronic Letters, Vol. 24(1), Jan. 1988, pp 6-7.

[3] Lee, K. F., Hon, H. W., and Reddy, R., (1990), "An Overview of the SPHINX Speech Recognition System," IEEE Trans. on ASSP, Vol. ASSP-38(1), Jan. 1990, pp. 35-45.

[4] Nishimura, M., and Toshioka, K., (1987), "HMM-Based Speech Recognition Using Multi-Dimensional Multi-Labeling," Proceedings of ICASSP-87, pp. 1163-1166.

[5] Rabiner, L. R., Levinson, S. E., & Sondhi, M. M., (1983), "On the Application of Vector Quantization and Hidden Markov Models to Speaker-Independent, Isolated Word recognition," AT&T Tech. J., Vol. 62(4), April 1983, pp. 1075-1105.

[6] Schwartz, R., *et al*, (1989), "Robust Smoothing Methods for Discrete Hidden Markov Models," Proceedings of ICASSP-89, pp. 548-551.

[7] Linde, Y., Buzo, A., and Gray, R. M., (1980), "An Algorithm for Vector Quantizer Design", IEEE Trans. on Comm., COM-28(1), 1980, pp. 84-95.

[8] Zhang, Y., deSilva, C., Attikiouzel Y., and Alder, M., (1992), "A HMM/EM Speaker-Independent Isolated Word Recognizer", The Journal of Electrical and Electronic Engineering, Australia, Vol. 12, No.4, Dec. 1992, pp. 334-340.

[9] Wolf, J. H., (1970), "Pattern Clustering by Multivariate Mixture Analysis", Multivariate Behavioural Research, Vol. 5, 1970, pp. 329-350.