Speaker Independent Isolated Digit Recognition Using Hidden Markov Models

S. E. Levinson
L. R. Rabiner
M. M. Sondhi

Acoustics Research Department
Bell Laboratories
Murray Hill, New Jersey 07974

ABSTRACT. A method for speaker independent isolated digit recognition based on modeling entire words as discrete probabilistic functions of a Markov chain is described. Training is a three part process comprising conventional methods of linear prediction coding (LPC) and vector quantization of the LPCs followed by an algorithm for estimating the parameters of a hidden Markov process. Recognition utilizes linear prediction and vector quantization steps prior to maximum likelihood classification based on the Viterbi algorithm. Vector quantization is performed by a K-means algorithm which finds a codebook of 64 prototypical vectors that minimize the distortion measure (Itakura distance) over the training set.

After training based on a 1,000 token set, recognition experiments were conducted on a separate 1,000 token test set obtained from the same talkers. In this test a 3.5% error rate was observed which is comparable to that measured in an identical test of an LPC/DTW (dynamic time warping) system. The computational demand for recognition under the new system is reduced by a factor of approximately 10 in both time and memory compared to that of the LPC/DTW system. It is also of interest that the classification errors made by the two systems are virtually disjoint; thus the possibility exists to obtain error rates near 1% by a combination of the methods.

In describing our experiments we discuss several issues of theoretical importance, namely: 1) Alternatives to the Baum-Welch algorithm for model parameter estimation, e.g., Lagrangian techniques; 2) Model combining techniques by means of a bi-partite graph matching algorithm providing improved model stability; 3) Methods for treating the finite training data problem by modifications to both the Baum-Welch algorithm and Lagrangian techniques; and 4) Use of non-ergodic Markov chains for isolated word recognition.

We note that the experiments reported here are the first in which a direct comparison is made between two conceptually different (i.e. parametric and non-parametric) methods of treating the non-stationarity problem in speech recognition by implicitly dividing the speech signal into quasi-stationary intervals.

I. Introduction

The non-stationarity of a speech signal poses a major difficulty in Automatic Speech Recognition (ASR). There are currently two conceptually different methods of treating non-stationarity based on implicit segmentation of the speech signal into quasi-stationary intervals. The two implicit methods are parametric and non-parametric in the sense of Patrick [4]. Much of our previous work in ASR [5] has been of the non-parametric variety the essence of which is the computation of a spectral distance measure between an unknown utterance and a labeled prototype. In this paper, we describe a parametric method for speaker-independent isolated digit recognition based on modeling entire words as discrete probabilistic functions of a Markov chain. The results of our recognition experiments are then compared with those from previous (i.e. non-parametric) experiments.

Throughout this paper we shall refer to the non-parametric system by the (unfortunately) popular name of its crucial process, dynamic time warping (DTW). The other system will be called (more properly) the Hidden Markov Model (HMM) system. The dichotomy between the two is illustrated in Figure 1.

In our work, both systems operate on the speech signal as represented in terms of linear prediction coefficients (LPC). For the DTW system, training consists of collecting a number, typically 100, of different talkers' utterances of each vocabulary word. The sets of utterances of each word are individually clustered, resulting in a smaller number, typically 12, of templates of each word. In the recognition mode, the distance between an unknown utterance and each template for each word is computed by a dynamic programming algorithm. The distances are entered into a nearest-neighbor decision rule on the basis of which a classification is made.

The HMM system is trained in two stages. First, the training data (exactly that presented to the DTW system) is used to select a quantization scheme for the LPC vectors. The quantized training data are then used to estimate the parameters of a single hidden Markov model for each word. Recognition is accomplished by quantizing the unknown input, computing the probability that it was generated by each word model and applying a maximum likelihood decision rule.

As indicated in Figure 1, training of the DTW system is essentially a data collection process which carries a small computational cost. Recognition under this system, however, requires a large number of distance computations and is thus computationally expensive. The apportionment of computational costs for the HMM system is exactly opposite. Training cost for

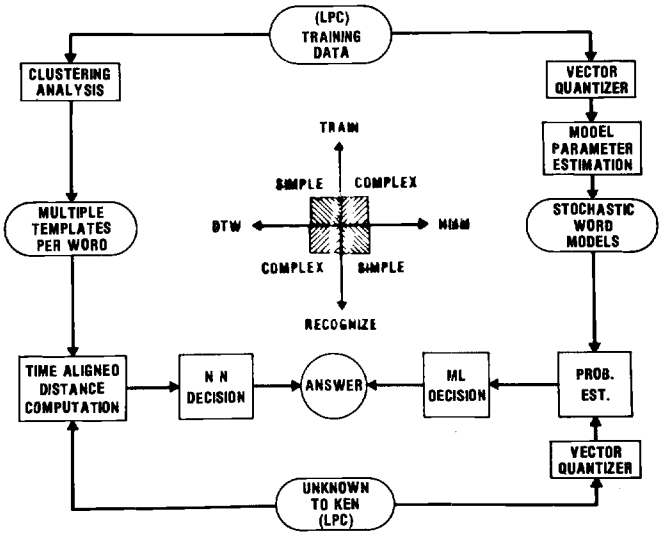

Figure 1 Block diagram of the Recognition Systems.

 CH1841-6/83/0000-1049 $1.00

the HMM is dominated by the parameter estimation process which is a complex constrained optimization problem. Recognition, on the other hand, requires but a simple algorithm to evaluate a set of likelihood functions. Here, then, the HMM system enjoys a distinct advantage in as much as its costly aspect need be performed only once, in training, while the DTW system requires the execution of its costly process with each recognition.

It should also be pointed out that the vector quantization phase is a convenience not strictly required by the theory underlying the HMM system. It is also true that the DTW system can accommodate a vector quantization stage. In fact, we report here the recognition results obtained with a vector-quantized DTW system.

The next four sections of this paper are devoted to a more detailed description of the components of the HMM system. We present, in order, the vector quantizer, the hidden Markov model, the parameter estimation procedure and the maximum likelihood classification procedure. Section VI describes our experimental procedure and the results thereof. Finally in Section VII we evaluate these results and discuss several related aspects of the computations.

To the best of our knowledge, this paper is the first to report a direct comparison of a DTW and an HMM system. Our implementations of these systems have achieved comparable error rates of less than 4% in a speaker independent isolated digit recognition task. The HMM system requires an order of magnitude less computation and storage than the DTW system in the recognition stage. Finally we have noted that errors made by both systems are detectable and disjoint thus allowing the possibility for construction of a robust hybrid system.

II. Vector Quantization

The simplest form of hidden Markov model is one in which the probabilistic function associated with each state can assume only one of a finite number of values. Since the LPC vectors extracted from the speech-signal are elements of an eight dimensional continuum, they must be quantized in order to be appropriate to the proposed model.

The vector quantization scheme which we employed for this purpose is a variant of the method of Juang et. al. [3] based on the well known K-means algorithm. Since we use the LPC representation of the speech signal, it is natural to adopt the Itakura [2] metric as a distortion measure. At the n^{th} stage of the process, the training data is clustered by the K-means algorithm into 2^n clusters the centers of which compose the n^{th} codebook. Within any given stage the K-means algorithm was iterated until the ratio of average distortions at two successive iterations fell below some predetermined threshold.

In our experiments, the training data consisted of one utterance of each of the ten digits spoken by 100 talkers (50 men and 50 women). The data comprises 39708 LPC vectors. Vector quantizers having codebooks of 2,4,8,16,32,64 and 128 entries were generated. For the 128 entry codebook, the average distortion was 0.165. Since the 64 entry codebook resulted in an average distortion of 0.22 which is significantly less than the mean of the theoretically predicted distribution of distortion of .37 (i.e. the mean value of an appropriately normalized χ^2 distribution with 8 degrees of freedom), we used the 64 entry vector quantizer in our experiments.

Figure 2 shows that the spectra corresponding to the entries of the 64-vector codebook give fairly uniform coverage of the vowel space of English. The graphs in Figure 2 were obtained by extracting the first three roots of the LPC polynomial. In part a, all

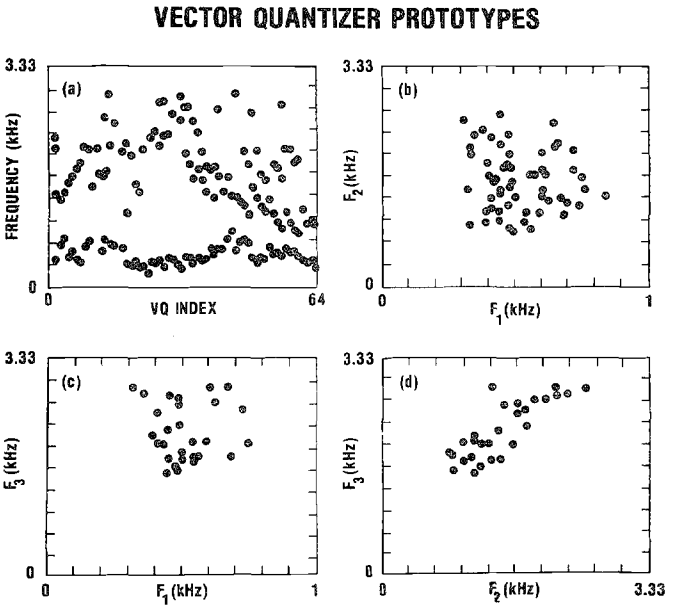

Figure 2 Spectra of the Vector Quantization Codebook.

roots are plotted as a function of quantizer index; parts b, c and d are scatter plots of all pairs of roots corresponding to each quantizer index.

Further evidence of the generality of the vector quantizer was obtained from synthesis experiments. Ordinary English sentences were synthesized using LPC coefficients both with and without vector quantization. The synthesis utilized monotone pitch and no voiced/unvoiced decision. Under these conditions, the synthetic utterances were perceptually indistinguishable.

III. The Hidden Markov Model

A probabilistic function of a (hidden) Markov chain is a stochastic process generated by two interrelated mechanisms: an underlying Markov chain having a finite number of states, and a set of random functions one of which is associated with each state. At discrete instants of time, the process is assumed to be in a unique state and an observation is generated by the random function corresponding to the current state. The underlying Markov chain then changes states according to its transition probability matrix. The observer sees only the output of the random functions associated with each state and cannot directly observe the states of the underlying Markov chain, hence the term hidden Markov model.

It is quite natural to think of the speech as being generated by such a process. We can imagine the vocal tract as being in one of a finite number of articulatory configurations or states. In each state a short (in time) signal is produced which has one of a finite number of prototypical spectra depending, of course, on the state. Thus the power spectra of short intervals of the speech signal are determined solely by the current state of the model while the variation of the spectral composition of the signal with time is governed predominantly by the probabilistic state transition law of the underlying Markov chain. For speech signals derived from a small vocabulary of isolated words, the model is reasonably faithful. The foregoing is, of course, an oversimplification intended only for the purpose of motivating the following discussion.

In our experiments, we used a 5-state, non-ergodic model with the special topology shown in Figure 3. The parameters of this model are the transition matrix, **A**, and the symbol probability matrix **B**. The symbols are the vector quantizer indices, v_k, $1 \le k \le 64$. Thus we can write $\mathbf{A} = [a_{ij}]$ where $a_{ij} = \text{Prob}(q_j \text{ at } t+1 | q_i \text{ at } t)$ for $1 \le i,j \le 5$ and $\mathbf{B} = [b_{jk}]$ where $b_{jk} = \text{Prob}(v_k \text{ at } t | q_j \text{ at } t)$ for $1 \le j \le 5$ and $1 \le k \le 64$.

From Figure 3 it is clear that the model must start in state q_1 and must end in state q_5. Furthermore, any state, once left, cannot be revisited. These constraints are imposed on the model in the training procedure by setting the initial estimates of the forbidden transitions to zero.

IV. Parameter Estimation

After vector quantization, our training data may be thought of as a long sequence, **O**, of observations for each word. Since the observations are derived from 100 independent utterances of the word, we write $\mathbf{O} = \mathbf{O}^{(1)}\mathbf{O}^{(2)}\cdots\mathbf{O}^{(100)}$. Each $\mathbf{O}^{(k)}$ is in turn a sequence, $\{O_t^{(k)}\}_{t=1}^{T_k}$ of symbols, vector quantizer indices. From Baum [1] it is clear that for the model of Figure 3, the probability, P_k, that $\mathbf{O}^{(k)}$ was generated by a model with parameters **A** and **B** is just

$$P_k = \text{Prob}(\mathbf{O}^{(k)}|\mathbf{A},\mathbf{B}) = \mathbf{e}_1\mathbf{B}_1^k\mathbf{A}\mathbf{B}_2^k\cdots\mathbf{A}\mathbf{B}_{T_k}^k\mathbf{e}_5' \qquad (1)$$

where $\mathbf{e}_1$ and $\mathbf{e}_5$ are, respectively, the first and fifth unit vectors signifying that the model starts in state q_1 and ends in state q_5 and $\mathbf{B}_t^k$ is a diagonal matrix whose j^{th} non-zero entry is $b_{j\ell}$ where $\ell = O_t^{(k)}$.

Since the utterances are assumed to be independent, the probability, **P**, of the entire training sequence is

$$\mathbf{P} = \text{Prob}(\mathbf{O}|\mathbf{A},\mathbf{B}) = \prod_{k=1}^{100} P_k \qquad (2)$$

Training is then a problem of finding model parameters, **A** and **B** so that **P** in (2) is maximized for the given training data **O**. This is a classical problem in constrained optimization. The constraints enter because **A** and **B** must be row-wise stochastic since their entries are probabilities.

The Baum-Welch algorithm [1] affords a particularly elegant method of maximizing **P**. Given estimates of the matrices **A** and **B**, we can form new ones $\bar{\mathbf{A}}$ and $\bar{\mathbf{B}}$ according to

$$\bar{a}_{ij} = \frac{a_{ij}\dfrac{\partial \mathbf{P}}{\partial a_{ij}}}{\displaystyle\sum_{k=1}^{5} a_{ik}\frac{\partial \mathbf{P}}{\partial a_{ik}}} \qquad 1 \le i,j \le 5 \qquad (3)$$

and

$$\bar{b}_{jk} = \frac{b_{jk}\dfrac{\partial \mathbf{P}}{\partial b_{jk}}}{\displaystyle\sum_{\ell=1}^{64} b_{j\ell}\frac{\partial \mathbf{P}}{\partial b_{j\ell}}} \qquad \begin{matrix} 1 \le j \le 5 \\ 1 \le k \le 64 \end{matrix} \qquad (4)$$

For each iteration of (3) and (4), $\text{Prob}(\mathbf{O}|\bar{\mathbf{A}},\bar{\mathbf{B}}) \ge \text{Prob}(\mathbf{O}|\mathbf{A},\mathbf{B})$ with equality iff $(\mathbf{A},\mathbf{B})$ is a critical point of **P**. Thus repeated applications of (3) and (4) starting from any initial estimate will often converge to a local maximum of **P**. Notice also that according to (3) and (4), a parameter initially estimated to be zero will remain zero in all subsequent estimates. Thus, the constraints required by our non-ergodic model can be easily imposed.

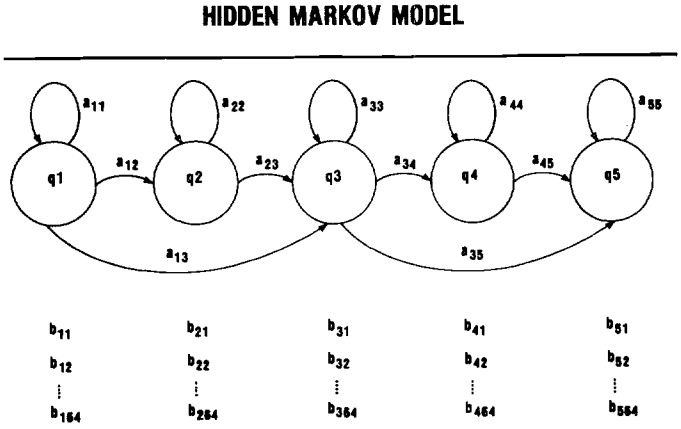

Figure 3 The Hidden Markov Model.

It is appropriate to note that the required optimization can also be performed by any of a number of classical non-linear programming algorithms based on Lagrangian techniques.

V. Classification

Our particular classification problem is the following. We wish to recognize isolated digits represented by $w_0, w_1, w_2, \ldots, w_9$. We are given an observation sequence **O** derived from the utterance of a digit and a set of models $(\mathbf{A}_0,\mathbf{B}_0)$, $(\mathbf{A}_1,\mathbf{B}_1)\cdots(\mathbf{A}_9,\mathbf{B}_9)$ determined by the method outlined in Section IV above. We compute $\mathbf{P}_i = \text{Prob}(\mathbf{O}|\mathbf{A}_i,\mathbf{B}_i)$ for $0 \le i \le 9$. The unknown utterance is then classified as w_i iff $\mathbf{P}_i > \mathbf{P}_j$ for $0 \le i,j \le 9$ and $j \ne i$.

The class conditional probabilities can be computed by direct evaluation of the likelihood function (1).

Alternatively we may take $\mathbf{P}_i$ to be the maximum over all possible state sequences $\mathbf{q} = q_0 q_1 q_2 \cdots q_T$ of the joint probability $\text{Prob}(\mathbf{O},\mathbf{q}|\mathbf{A}_i,\mathbf{B}_i)$. This distinguished state sequence and the probability of its corresponding observation sequence can be simultaneously computed with the Viterbi [6] algorithm as follows. Let $\phi_1(j) = b_{j\ell}$, $\ell = O_1$ and $j = 1$; $\phi_1(j) = 0$ for $2 \le j \le 5$. Then for $2 \le t \le T$, and $1 \le j \le 5$

$$\phi_t(j) = \max_{[1\le\ell\le 5]} \left\{\phi_{t-1}(\ell)a_{\ell j}^{(i)}b_{jk}^{(i)}\right\}; \qquad k = O_t \qquad (5)$$

after which, $\mathbf{P}_i = \phi_T(5)$ where the superscript i indicates that the parameters are those of the i^{th} model.

While direct evaluation of the likelihood function from (1) and computation based on (5) will yield different values for $\mathbf{P}_i$, we have found that they yield identical classifications in the digit task.

VI. Experimental Results

The data base for our experiments consisted of two separate sets of 1000 utterances, one sample of each of ten digits uttered by each of 100 subjects, 50 men and 50 women. The talkers were the same for both the training and test sets; data collection for the two sets took place several weeks apart. The subjects were all native speakers of American English.

The training set was used both to select the vector quantization codebook and to estimate the parameters of the Markov models. After these processes were completed no further use was made of the training data.

The test set was then used to conduct three recognition experiments. The 1000 utterances were recognized with the DTW and HMM systems described in Figure 1 and with a vector quantized version of the DTW system. The results are summarized in Table 1. The error rates are all less than 4%; we do not consider the variation in error rates significant enough to recommend one system over another. The HMM system does, however, enjoy a distinct advantage in computational complexity requiring an order of magnitude less storage and time for execution.

The errors made by the DTW and HMM systems were found to be detectable and disjoint. Whenever the first and second candidates under the DTW system had metrics separated by less than a fixed threshold, a misclassification was likely. In such cases, the HMM system usually yielded a correct result. Thus a hybrid system with error rate near 1% is possible.

VII. Conclusion

In addition to the experimental results given above, we made some observations of theoretical interest. Space limitations allow us to do no more than list them here. As we noted in Section IV, the optimization operation necessary for model parameter estimation can be carried out by classical methods. These methods which rely on the gradient and possibly higher order derivatives seem to have some advantages, both in speed and generality, over the Baum-Welch algorithm.

Very often, insufficient training data will result in zero values for certain parameters. These zeros can be fatal to classification. It is, however, possible to perform the parameter estimation so that the parameters are constrained to be greater than zero. Such constraints can be accommodated either by the Baum-Welch or Lagrangian techniques.

Under certain circumstances it may be useful to combine one or more models. An obstacle to doing so is that the states of two or more models may be permutations of each other. Combination cannot be accomplished until the corresponding states of each model are known. An optimal isomorphism between models can be efficiently found by means of a bi-partite graph matching algorithm.

In conclusion, we note that the experiments reported here are the first in which a direct comparison is made between DTW and HMM types of systems. These experiments show that the systems provide comparable accuracy but make different errors. Further study is required to understand the significance of these observations.

References

[1] Baum, L. E., "An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of a Markov Process," *Inequalities*, 3, 1-8, 1972.

[2] Itakura, F., "Minimum Prediction Residual Principle Applied to Speech Recognition," *IEEE Trans. on Acoust. Speech and Signal Processing*, ASSP-23 (1), 67-72, 1975.

[3] Juang, B. H., Wong, D. Y. and Gray, A. H. Jr., "Distortion Performance of Vector Quantization for LPC Voice Coding," *IEEE Trans. on Acoust. Speech and Signal Processing*, ASSP-30 (2), 294-304, 1982.

[4] Patrick, E. A., *Fundamentals of Pattern Recognition*, Prentice-Hall, Englewood Cliffs, NJ, 1972.

[5] Rabiner, L. R., Levinson, S. E., Rosenberg, A. E. and Wilpon, J. G., "Speaker Independent Recognition of Isolated Words Using Clustering Techniques," *IEEE Trans. on Acoust. Speech and Signal Processing*, ASSP-27 (4), 336-49, 1979.

[6] Viterbi, A. J., "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm," *IEEE Trans. Information Theory*, IT-13 (2), 260-9, 1967.

TABLE I—COMPARISON OF RESULTS ON HMM/VQ AND LPC/DTW WORD RECOGNIZERS

	RECOGNIZER		
DIGIT	HMM/VQ	LPC/DTW	LPC/DTW/VQ
0	98	99	99
1	98	98	99
2	96	100	96
3	99	99	97
4	93	97	96
5	97	96	93
6	96	100	94
7	99	100	94
8	92	98	96
9	95	98	96
AVERAGE	96.3	98.5	96.5

RESULTS (AVERAGE WORD ACCURACY (%))
100 TALKERS, 10 DIGITS PER TALKER