S6.1

Application of Hidden Markov Models for Recognition of a Limited Set of Words in Unconstrained Speech

J. G. Wilpon, C. H. Lee and L. R. Rabiner
AT&T Bell Laboratories
Murray Hill, New Jersey 07974

Abstract. Speaker independent recognition of small vocabularies, spoken over the long distance telephone network, has been demonstrated to be a viable technology. For most tasks it is generally assumed that users will be cooperative and only speak the predefined vocabulary words in isolation. Recently, a large scale trial of speaker independent isolated word speech recognition technology was carried out in Hayward, California. The task chosen required that users speak, in isolation, one of five defined vocabulary words (*collect, calling-card, person, third-number* and *operator*). At ICASSP-88 [1], we presented recognition results which showed that when users spoke the vocabulary words in isolation, they were correctly recognized about 99% of the time. However, observation of customer responses during this trial indicated that about 20% of the utterances had the desired vocabulary item along with extraneous input which ranged from non-speech sounds to groups of non-vocabulary words (e.g. 'I want to make a *collect* call please'). Most recognition algorithms have not been designed to handle this type of input. As such, a modification of the algorithms had to be made to recognize selected vocabulary words embedded in speech (i.e. a form of keyword spotting). For our task, it was assumed that for each spoken input, which consisted of a period of background signal, speech, and another period of background signal, a single one of the keywords was spoken. As such, this form of word spotting is a significantly easier task than is usually associated with keyword spotting in fluent speech. Recognition results on the keywords embedded in speech were 87.1%. To achieve this performance we used several robust training methods as well as detailed analyses of state-level likelihood and duration scores.

1. Introduction

The development of robust, speaker-independent, speech recognition systems that perform well over dialed-up telephones line has been a topic of interest for over a decade [1,4-7]. This work has progressed from systems that can recognize a small number of vocabulary items spoken in isolation [3,5,7], to systems that can recognize medium size vocabulary sets spoken fluently, [4]. A basic assumption for most speech recognition systems is that the input to be recognized consists solely of words from the recognition vocabulary and background silence. However, previous studies on the recognition of a limited set of isolated command phrases for making "operator assisted calls" have shown that it is extremely difficult, if not impossible, to get real-world subscribers to such a service to speak only the allowed input words [1,5,6]. In a large scale trial of speaker independent, isolated word, speech recognition technology, carried out at an AT&T central office in Hayward, California (i.e. the San Francisco Bay area) [1,8], live telephone customer traffic was used to evaluate the call handling procedures being developed for a new generation of telephone switching equipment. Using these procedures, customers making operator assisted calls were given the option of verbally identifying the type of call they wished to make (i.e. *collect, calling-card, person-to-person, bill-to-third,* and *operator*). Each caller was requested to speak one of the five orally prompted commands in an isolated fashion. While 82% of the users actually spoke one of the command words, only 79% of these inputs were spoken in isolation (i.e. only 65% of all the callers followed the protocol). Monitoring of the customer's spoken responses indicated that 17% of all responses contained a valid vocabulary item along with extraneous speech input (e.g. *I want to make a* collect *call please*). Most conventional isolated word recognition algorithms have not been designed to recognize vocabulary items embedded in various carrier sentences. As such, modifications to the algorithms had to be made to allow for the recognition of words embedded in speech (i.e. a form of keyword spotting).

In this paper we discuss the problem of recognizing a small set of words spoken in the context of unconstrained input. The task is significantly easier than the general case of keyword spotting in fluent speech. In the general case, the spotting system is presented with continuous input and must make a decision whether or not any of the keywords is present anywhere in the speech. For our task, it is known that for each spoken input a single one of the keywords has been spoken.

While much research has been performed on the general wordspotting task, very little of it has been published. Most of the techniques that have been described in the literature are template-based dynamic time-warping approaches (DTW) [9-11]. In [9], Christiansen and Rushforth describe a speaker trained keyword spotting system which uses an LPC representation of the speech signal without any syntactic or semantic information about the task. Using this approach they achieved good results on a vocabulary set of four words and the ten digits. Myers *et al* [11] discussed an approach which used a local minimum DTW-based algorithm for the problem of word spotting. However the proposed system was not evaluated on any real task. Higgins and Wohlford [10], also proposed a DTW-based system for keyword spotting. In their system, knowledge about the vocabulary and syntax of the input speech was used. A set of keyword templates and non-keyword templates were created and compared against several pooled *filler* templates (where *filler* templates were created from clustering a large set of non-vocabulary word speech utterances) as to their ability to detect keywords in fluent speech. Their results indicated that while explicit knowledge of the vocabulary may not be that important, the use of *filler* templates may be important.

A significant amount of progress has been recently made in automatic speech recognition using hidden Markov modeling (HMM) [12-15]. Since the HMM approach uses a statistical parameterization of the signal, it should contain more information about the signal than the DTW-based approach. As such, we chose to develop an algorithm using HMM technology to attack the problem of recognizing a small set of vocabulary words in fluent speech.

2. HMM-Based Recognition Algorithm

Figure 1 shows a block diagram of the HMM-based recognition system. The key elements of the system include:

2.1 LPC and Cepstral Analysis

Speech is first digitized at a 6.67 kHz rate and filtered to a bandwidth of 200 - 3200 Hz. The digitized speech is then preemphasized using a simple first-order digital filter with a preemphasis factor $a = 0.95$, and blocked into frames of 45 msec in length with a shift between frames of 15 msec. Each frame of speech is weighted by a Hamming window. An 8-th order linear predictive coding (LPC) analysis is then performed on the data. Thus, for each frame, a set of eight LPC coefficients is generated. The input signal is then reduced to a sequence of LPC frame vectors. There is no automatic endpoint detection performed on the data. The LPC derived cepstral vector is then computed up to the Q^{th} component, where $Q > p$, and $Q = 12$ in our implementation.

The Q-coefficient cepstral vector, $c_\ell(m)$, at time frame ℓ is weighted by a window, $W_c(m)$, of the form:

$$W_c(m) = \left[1 + \frac{Q}{2}\sin\left(\frac{\pi m}{Q}\right)\right], \quad 1 \le m \le Q \qquad (1)$$

to give:

$$\hat{c}_\ell(m) = c_\ell(m) \cdot W_c(m) \qquad (2)$$

It has recently been shown that by extending the analysis vector to include spectral derivative (in time) information, performance of

CH2673-2/89/0000-0254 $1.00 © 1989 IEEE

several standard speech recognizers improved significantly [13]. As such we include such spectral derivative information in our analysis vector as follows.

The time derivative of the sequence of weighted cepstral vectors is approximated by a first order orthogonal polynomial over a finite length window of $(2K + 1)$ frames, centered around the current vector. ($K = 2$ in our implementation; hence the derivative is computed from a 5 frame (75 msec) window.) The cepstral derivative (i.e. the delta cepstrum vector) is computed as

$$\Delta \hat{c}_\ell(m) = \left[\sum_{k=-K}^{K} k\, \hat{c}_{\ell-k}(m)\right] \cdot G, \quad 1 \le m \le Q \tag{3}$$

where G is a gain term so that the variances of $\hat{c}_\ell(m)$ and $\Delta \hat{c}_\ell(m)$ are about the same. (For our system the value of G was 0.375.)

The overall observation vector, $\mathbf{O}_\ell$, used for scoring the HMM's is the concatenation of the weighted cepstral vector, and the corresponding weighted delta cepstrum vector, i.e.

$$\mathbf{O}_\ell = \left\{\hat{c}_\ell(m),\ \Delta \hat{c}_\ell(m)\right\} \tag{4}$$

and consists of 24 coefficients per vector.

2.2 Structure of Hidden Markov Models

Figure 2 illustrates the structure of the HMM's used to characterize individual words as well as the background noise. The models are first order, left-to-right, Markov models with N states. Each model consists of the following parameters:

(1) a state transition matrix, $A = a_{ij}$ with the constraint that

$$a_{ij} = 0 \quad j < i,\ j \ge i + 2 \tag{5}$$

(i.e. we allow transitions from state j only to itself, or to state $j + 1$).

(2) a continuous mixture density matrix $B = b_j(\mathbf{x})$ of the form

$$b_j(\mathbf{x}) = \sum_{m=1}^{M} c_{mj} N[\mathbf{x},\ \mu_{mj},\ \mathbf{U}_{mj}] \tag{6}$$

where $\mathbf{x}$ is the input cepstral vector, c_{mj} is the mixture weight for the mth component in state j, μ_{mj} is the mean vector for mixture m in state j, and $\mathbf{U}_{mj}$ is the covariance for mixture m in state j. All evaluations described in this paper used diagonal covariance matrices. In our evaluations, the number of states per model was set to 10 and the number of mixture components per state, M, was set to nine. (Several other values for N and M were evaluated.)

(3) a set of log energy densities, $p_j(\epsilon)$, where ϵ is the dynamically normalized frame energy, and p_j is an empirically measured discrete density of energy values in state j.

(4) a set of state duration probabilities, $\hat{p}_j(\tau)$, where τ is the number of frames spent in state j, and $\hat{p}_j$ is an empirically measured discrete density of duration values in state j.

2.3 Model Alignment Procedure

A sequence of spectral vectors for the unknown speech signal is matched against a set of stored reference models (hidden Markov models) using a Viterbi algorithm. This matching occurs at each frame of the input signal, thereby generating a sequence of best matches for each frame. This approach, similar to the one used in the template-based system described by Christiansen and Rushforth [9], can be thought of as sliding the input speech past each model in a continuous manner. The HMM-based approach uses a frame-synchronous Viterbi decoding procedure, described by Lee and Rabiner [14], to perform the time alignment. For every beginning frame, i, a Viterbi alignment is generated for every possible ending frame j to produce a candidate $c(i, j)$ based on the best aligned model likelihood. The candidate $c(i, j+1)$ is obtained frame synchronously using the alignment information generated in producing $c(i, j)$ plus the local likelihood measure at frame $j+1$.

For every pair of beginning and ending frames, i and j, the alignment procedure produces a word candidate $c(i, j)$, with average model likelihood $p(i, j)$ and average state likelihood $s(i, j)$ defined as follows:

$$p(i, j) = \frac{1}{j-i+1} \sum_{k=i}^{j} m_p(k) \tag{7}$$

where $m_p(k)$ is the aligned likelihood score at frame k for candidate $c(i, j)$, and

$$s(i, j) = \frac{1}{N} \sum_{n=1}^{N} s_p(n) \tag{8}$$

where $s_p(n)$ is the aligned average state likelihood in state n, for candidate $c(i, j)$.

Because of the exponential state duration constraints inherent in the HMM formulation, there is little limitation as to how long a portion of an utterance can stay in a particular state. During the Viterbi alignment process, it is possible for the spectrum of the input utterance to match the spectral density in some of the states in the model well and match the spectral density in other states poorly. As a result, an alignment path may remain in one or more states for a very long duration (with a very good likelihood score) and stay a realively short period of time in the remaining states (with a very bad likelihood score). When this happens the total likelihood score for the entire utterance will be high, but the average state likelihood will be low.

The outputs from the model alignment process, $c(i, j)$, $p(i, j)$, and $s(i, j)$, are then used in the postprocessor, which tests the output sequence for valid candidate words.

2.4 Postprocessor

The output candidate sequence is subjected to validity testing to eliminate unlikely candidates. The postprocessor chooses the most likely word from the remaining candidates. The following tests are made (in the order specified):

1. Duration test - The duration of each candidate must be within a predefined range. In our tests we set the minimum duration to 25 frames (375 msec) and left the maximum duration unspecified.
2. Energy level test - The log energy of the utterance is first normalized so that the minimum log energy is 0 dB. The maximum normalized energy level within the boundaries of the candidate word at frame i must be greater than some predefined threshold (in our tests the threshold was set to 30 dB). This test serves to eliminate word matches to background signal.
3. Average model likelihood test - The likelihood measure associated with the candidate word at frame i must exceed some predefined value. This threshold was set to 5.0 in our experiments. Typical model likelihood scores generally fall in the range 5.0-20.0 for the training data.
4. Ratio of average model likelihood to average state likelihood - The ratio of the average model likelihood to the average state likelihood was tested to see if it fell in the range

$$0.65 \le \text{ratio} \le 1.35 \tag{9}$$

If it fell outside this range the candidate word was eliminated from consideration. This occurred when the model match in a small number of the word states was very good, and the model match in the remaining states was very bad. Such matches were considered unreasonable.

If, after applying these tests, more than one candidate still remained the recognized word was chosen as the candidate with the highest average state likelihood score. If no candidates remained after postprocessing, the utterance was rejected and no decision was made. no recognition decision will be made.

2.4.1 Examples of Recognition Output

Figure 3 shows a plot of the recognition output from the model alignment procedure for the utterance: *Uh,* **calling-card** *please*, where the word **calling-card** is the keyword to be recognized. In Figure 3a, the best total likelihood score (solid line) for each frame of the input signal is shown along with the average state likelihood score (dotted line). Figure 3b shows a normalized log-energy contour and Figure 3c shows the top recognition candidate at each frame of the input. As discussed in the previous section, we see that very good scores are generally obtained in the background silence regions of the recording interval. However, the average state likelihood scores in the background region are much lower than the total likelihood scores.

For example, the total likelihood scores for frames 125 through 150 are much higher than those computed during the speech interval. Figure 4 shows similar plots after applying the postprocessing rules. (The horizontal lines in Figure 4b indicate the region where the recognized word was detected (spotted)). These figures show that, after postprocessing, only two valid candidates remain. A final recognition choice of the word **calling-card** was made because of its higher average state likelihood score.

3. Experimental Database

A speech database, consisting of approximately 75000 utterances, was collected during a large scale trial of speaker independent isolated word recognition technology, carried out at an AT&T central office in Hayward, California. The five word vocabulary defined for this task was, *collect*, *calling-card*, *third-number*, *person* and *operator*. Each utterance was obtained from a telephone customer during the course of a normal operator assistance call. Each caller was automatically prompted (by a voice response system) to speak one of the five keywords in an isolated fashion. During the trial about 17% of all customer responses contained a valid vocabulary word along with extraneous input which ranged from non-speech sounds, such as background music or TV sounds, to groups of words outside of the vocabulary, e.g. *I want to make a* **collect** *call please*.

Of the 75000 collected utterances, 7981 were digitized and used for experimentation. Table 1 shows the distribution of the digitized utterances that were used in our recognition experiments. Each customer's utterance contains only one of the vocabulary items. Roughly, half the utterances in each category were used for training; the other half were used for testing.

4. Recognition Results

Several different word HMM's were created and tested. Since extracting keyword tokens from continuous speech is a long and tedious job, one set of hidden Markov models (one model per word) was trained (using a segmental *k*-means training algorithm [15]) using only the isolated utterance data. The problem with this type of word training is that it does not account for any the coarticulation effects nor any of the word duration shortening effects that occur when the words are embedded in connected speech. Therefore, word models were also generated from word data excised by a trained listener from the connected portion of the database. (These tokens were also used to create one HMM per word). Additionally, a third set of word HMM's was created by pooling together the isolated word and the embedded word data. Table 1 shows that 3283 tokens were used to train the first model set and only 755 tokens were used to train the second model set.

Table 2 presents the results of a series of recognition tests performed on the portion of data where a keyword was spoken along with extraneous input. The results using the isolated utterance training database show that 87.2% of the utterances were correctly recognized with a 10.3% error rate and a 2.4% rejection rate. The recognition accuracy using the embedded utterance training set was 86.1% correct rate with 12.8% errors and 1.1% rejections. This result compares favorably with the previous result, since the training set was only about one quarter the size. The recognition accuracy obtained using models trained from the 4038 token combined database (86.7%) was slightly lower than that obtained using each of the two training data sets individually.

In Reference 12, Wilpon and Rabiner showed that explicit endpoint detection could be removed entirely from the recognition system while maintaining high recognition accuracy. To achieve this, the recognition system modeled the incoming signal as a sequence of *background signal* and *vocabulary words*. In ICASSP-88 [1], the portion of the database containing only isolated word input was tested on this algorithm and achieved a word accuracy rate of 98.9%. It was initially thought that this approach would extend naturally to the case of vocabulary words embedded in fluent speech, if a good model for the extraneous speech could be obtained. However, when this algorithm was tested on the embedded portion of the database, using the model for background noise described in [12], it achieved a word recognition rate of only 64.3% (as compared to the 87.1% recognition rate discussed previously). Therefore, the error rate on the entire database was 7.7% with no rejections. Using the keyword spotting algorithm described in Section 3, a recognition rate of 94.0% (with a 4.0% error rate and a 2.0% rejection rate) was obtained on the isolated portion of the database. Therefore, for the entire isolated and embedded speech database, 4.9% of the utterances were misrecognized with a 2.3% rejection rate.

The results achieved in these keyword spotting experiments are encouraging. However, some of the robustness issues related to the HMM formulation must be carefully investigated in order to obtain better performance. We have found the use of word duration, energy, and average state likelihood to be helpful in reducing the number of possible false alarms. We have found that the use of average state likelihood also improves the overall recognition performance. It should be possible to incorporate more state-level constraints, e.g. as minimum and maximum state duration, into the recognizer. Additionally, we are not currently using any information about the syntactic or semantic constraints imposed by the recognition task (as was suggested by Higgins and Wohlford [9]).

5. Conclusion

In this paper we have presented an algorithm based on hidden Markov models which can recognize a pre-defined set of vocabulary items spoken in the context of fluent speech. We have shown that, for a vocabulary of five words, we can correctly recognize 87.1% of keywords when they occur in fluent speech and spoken over the long distance telephone network. While the task that we are concerned with is significantly easier than what is normally associated with keyword spotting in continuous speech, it does address an important problem that must be solved for successful deployment of speech recognition technology.

References:

[1] Wilpon, J. G., DeMarco, D. M. and Mikkilineni, R. P., "Isolated word recognition over the DDD telephone network – results of two extensive field studies," *Conf. Rec. IEEE Int. Conf. Acous., Speech, and Sig. Processing*, **1S.1.10**, pp. 55 - 57, New York City, New York, April, 1988.

[2] Lee, Chin-Hui, "Some techniques for creating robust stochastic models for speech recognition," *Journ. of the Acoustical Society of America*, Supplement 1, Vol. 82, Fall 1987.

[3] Itakura, F., "Minimum Prediction Residual Principle Applied to Speech Applications," *IEEE Trans. on ASSP*, Vol. ASSP-23, No. 1, pp. 67-72, February 1975.

[4] Rabiner, L. R. and Levinson, S. E., "Isolated and Connected Word Recognition – Theory and Selected Applications," *IEEE Trans. on Comm.*, Vol. COM-29, No. 5, pp. 621-659, May 1981.

[5] Wilpon, J. G. and Rabiner, L. R., "On the Recognition of Isolated Digits from a Large Telephone Customer Population," *B.S.T.J.*, Vol 61, No. 7, pp. 1977-2000, September 1983.

[6] Wilpon, J. G., "A Study on the Ability to Automatically Recognize Telephone Quality Speech from Large Customer Populations," *AT&T Tech. J.*, Vol. 64, No. 2, pp. 423-451, February 1985.

[7] Rabiner, L. R. and Wilpon, J. G., "Some Performance Benchmarks for Isolated Word, Speech Recognition Systems," *Computer Speech and Language*, Vol. 2, No. 3/4, pp. 343-358, December, 1987.

[8] Thanawala, R., Fetz, B. H. and Piereth, R. J., "Automatic Speech Recognition in the Public Switch Network," *Proc. 5th World Telecom Forum*, Vol. 1, Part 2, pp. 235-238.

[9] Christiansen, R. W. and Rushforth, C. K., "Detecting and Locating Key Words in Continuous Speech Using Linear Predictive Coding," *IEEE Trans. on Acoustics, Speech and Signal Processing*, Vol. ASSP-25, No. 5, pp. 361-367, October, 1977.

[10] Higgins, A. L. and Wohlford, R. E., "Keyword Recognition Using Template Concatenation," *Conf. Rec. IEEE Int. Conf. Acous., Speech, and Sig. Processing*, pp. 1233-1236, Tampa, Florida, March, 1985.

[11] Myers, C. S., Rabiner, L. R. and Rosenberg, A. E., "An Investigation of the Use of Dynamic Time Warping for Word Spotting and Connected Word Recognition," *Conf. Acous. Speech and Sig. Processing*, pp. 173-177, Denver, CO, April 1980.

[12] Wilpon, J. G. and Rabiner, L. R., "Application of Hidden Markov Models to Automatic Speech Endpoint Detection," *Computer Speech and Language*, Vol. 2, No. 3/4, pp. 321-341, December, 1987.

[13] Rabiner, L. R., Wilpon, J. G. and Soong, F. K., "High Performance Connected Digit Recognition Using Hidden Markov Models," *Conf. Rec. IEEE Int. Conf. Acous. Speech and Sig. Processing*, Vol. 1, pp. 119-122, New York, NY, April 1988.

[14] Lee, C. H. and Rabiner, L. R., "A Network-Based Frame Synchronous Level Building Algorithm for Connected Word Recognition," *Conf. Rec. IEEE Int. Conf. Acous. Speech and Sig. Processing*, Vol. 1, pp. 410-413, New York, NY, April 1988.

[15] Rabiner, L. R., Wilpon, J. G. and Juang, B. H., "A Segmental *k*-means Training Procedure for Connected Word Recognition Based on Whole Word Reference Patterns," *AT&T Tech. Journ.*, Vol. 65, No. 3, pp. 21-31, May 1986.

Vocabulary Word	Total	Words Spoken as Isolated Utterances		Words Spoken as Part of a Connected Phrase	
		Training	Testing	Training	Testing
Collect	4320	1703	1602	510	505
Calling-card	2121	910	955	133	123
Third-number	619	273	281	34	31
Person	170	77	67	14	12
Operator	751	320	309	65	57
Totals	7981	3283	3214	755	728

TABLE 1

Hayward Speech Database Distributions

Training Data	Total Number of Tokens in Training Set	Recognition Accuracy		
		% Correct	% Error	% Rejected
(a) isolated tokens	3283	87.2	10.3	2.4
(b) embedded tokens	755	86.1	12.8	1.1
(c) isolated & embedded tokens	4038	86.7	11.8	1.5

TABLE 2

Recognition Results For Data Where A Vocabulary Word Was Spoken Along With Extraneous Speech

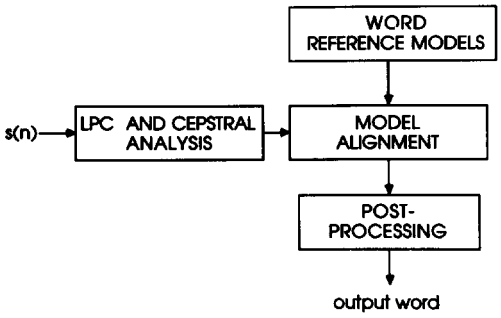

FIGURE 1
Block Diagram of Recognition System

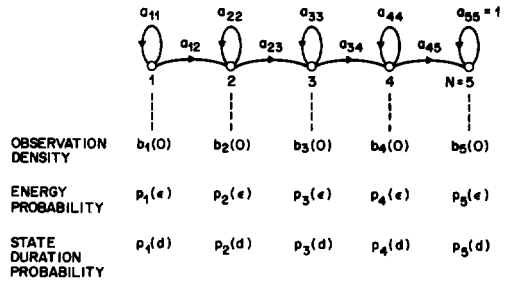

FIGURE 2
Structure of HMM Used to Characterize Individual Words

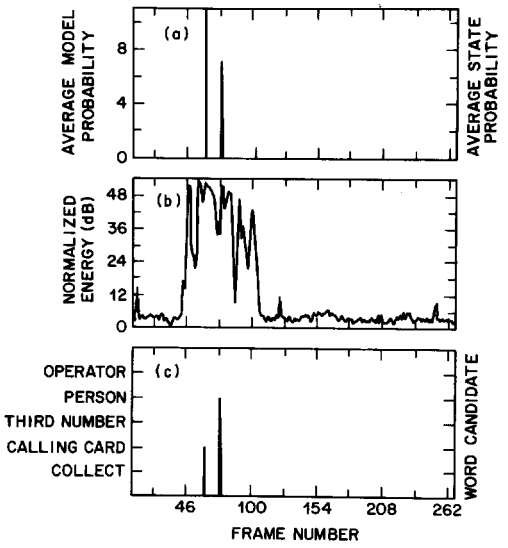

FIGURE 3
Output of Recognition System after Model Alignment Stage

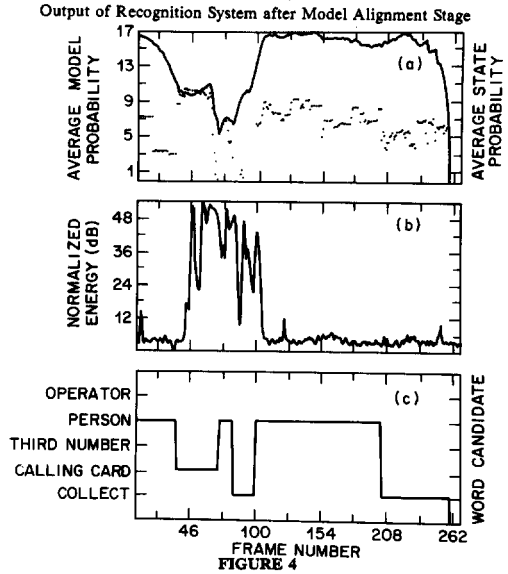

FIGURE 4
Output of Recognition System after Post-Processing Stage