ISOLATED WORD RECOGNITION USING THE HMM STRUCTURE SELECTED BY THE GENETIC ALGORITHM

Tomio Takara, Kazuya Higa, and Itaru Nagayama

Department of Information Engineering, University of the Ryukyus
1 Senbaru, Nishihara, Okinawa 903-01 JAPAN
takara@ie.u-ryukyu.ac.jp

ABSTRACT

Hidden Markov models (HMMs) are widely used for automatic speech recognition because they have a powerful algorithm used in estimating the models' parameters, and achieve a high performance. Once a structure of the model is given, the model's parameters are obtained automatically by feeding training data. There is, however, no effective design method leading to an optimal structure of HMMs. In this paper, we propose a new application of a genetic algorithm to search out such an optimal structure. In this method, the left-right structures are adopted for HMMs and the likelihood is used for the fitness of the genetic algorithm. We report the results of our experiment showing the effectiveness of the genetic algorithm in automatic speech recognition.

1. INTRODUCTION

Hidden Markov models (HMMs)[1] are widely used for automatic speech recognition because they have a powerful algorithm used in estimating the models' parameters, and achieve a high performance. Once a structure of the model is given, the model's parameters are obtained automatically by feeding training data. However, there is a problem still unresolved, i.e. how to design the optimal structure of an HMM.

One of the answers to this problem is the successive state splitting algorithm[2]. However the resulting structure of this method may not be optimal because the structure is searched locally. In order to search out the optimal structure, a wider scope search is needed.

One of the effective methods for a wide scope search is the genetic algorithm(GA)[3]. In this algorithm, a candidate for the solution of a problem is represented by a one dimensional string of genotype on a chromosome. The string is decoded into a phenotype and its fitness is evaluated. Individuals with higher fitness survive and individuals with lower fitness die. Finally, the optimal solution with the highest fitness is obtained.

The GA was applied to search out the optimal structure of multi-state Markov models (MSMMs) for automatic speech recognition[4]. However, the layered structure of an MSMM is restricted, and cannot efficiently express variant structures of the Markov model. The flexible structure of an HMM can express more variants than the layered structure of an MSMM. The GA has also been applied to select the HMM structure for DNA signal pattern extraction[5]. This method, however, is only applied to a two-class problem of a pattern recognition, and can not be applied as is to automatic speech recognition because speech recognition is inherently a multi-class problem.

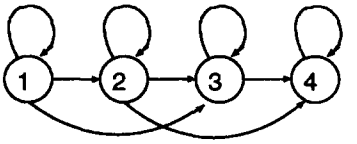

Figure 1: An example of HMM structure.

In this paper, we apply the GA to automatic speech recognition, in which the HMM structure is optimized for each word class. One of the initial individual structures has a simple left-right form in which no state-transition jumps states. We also discuss the effects of elite-preservation and fitness-ordered strategies of the GA in automatic speech recognition.

2. SPEECH RECOGNITION USING HIDDEN MARKOV MODELS

A hidden Markov model (HMM) is understood as a generator of vector sequences, and has a number of states connected by arcs. Figure 1 illustrates an example of an HMM structure, in which the circles and the arrow arcs represent the states and the state-transitions, respectively. In each state, there is an output probability distribution of an acoustic vector, and each transition is associated with a state-transition probability. These probabilities are called the model parameters and can be estimated effectively by using the Baum-Welch algorithm[1]. An HMM structure can be expressed in a matrix form $C = (c_{i,j})$. When $c_{i,j} = 1$, there exists a transition from state i to state j, and when $c_{i,j} = 0$, the transition does not exist. For example, the matrix expression of the structure of Figure 1 is shown in Figure 2. The matrix expression of an HMM will be used for the coding of the genetic algorithm.

An HMM is a finite-state machine that changes state once every time unit. Each time, t, a state, j, is entered, an acoustic speech vector, $\mathbf{y}_t$, is generated with probability density $b_j(\mathbf{y}_t)$. The transition from state i to state j is

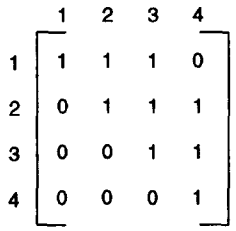

Figure 2: Matrix expression of Figure 1.

governed by the probability $a_{i,j}$. The joint probability of a vector sequence $\mathbf{Y}$ and state sequence $\mathbf{X}$, given some model M is calculated as the product of the transition probabilities and the output probabilities:

$$p(\mathbf{Y}, \mathbf{X}|M) = a_{x(0),x(1)} \prod_{t=1}^{T} b_{x(t)}(\mathbf{y}_t) a_{x(t),x(t+1)} \quad (1)$$

where $x(0)$ is constrained to be the model entry state and $x(T+1)$ is constrained to be the model exit state. Eq.(1) can be rewritten in a logarithmic form $\log p(\mathbf{Y}, \mathbf{X}|M)$. Using the Viterbi algorithm, $\log p(\mathbf{Y}|M)$ can be approximated by finding the state sequence $\mathbf{X}$ that maximizes Eq.(1). We adopt the Viterbi algorithm to calculate the log-likelihood, $\log p(\mathbf{Y}|M)$.

In a spoken word recognition system using HMMs, the HMMs for each word class are previously prepared. When a spoken word is inputted, the log-likelihoods for each HMM of a word class are calculated and the word class maximizing this value is determined as the word class of the inputted word.

We also use the log-likelihood to evaluate the fitness of the genetic algorithm. In this case, the evaluations are done for each training datum and are averaged in the word class. In order to prevent being affected by a word length, in our method, the log-likelihood of a word is divided by the word length T.

3. SELECTION OF AN HMM STRUCTURE USING THE GENETIC ALGORITHM

The genetic algorithm (GA) was introduced on the basis of the principle of biological evolution (natural selection and mutation) and has been used for search, training or optimization. In this algorithm, a candidate for the solution of a problem is represented by a one dimensional string of genotype on a chromosome. The string is decoded into a phenotype and its fitness is evaluated. Individuals with higher fitness survive and individuals with lower fitness die. The procedure of the GA is as follows:

1 Set initial generation.

2 Repeat following GA operations until the terminating condition is satisfied.

- Fitness evaluation
- Selection
- Crossover
- Mutation

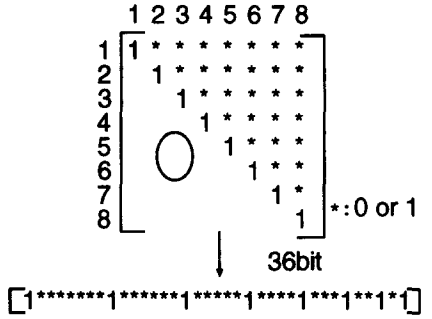

Figure 3: The coding for the HMM structure.

We apply the GA to search for the optimal HMM structure. We adopt the left-right (L-R) type HMM structure because with it, we can associate time with the model states in a straightforward manner[1]. For simplicity, we set the number of states to be constant in all generations. The L-R HMM structure represented by the matrix form is coded into the genotype string as shown in Figure 3.

We adopt log-likelihood as the fitness. The fitness evaluation in our method is done as follows: First, the model parameters of an HMM structure are randomly initialized and estimated by the Baum-Welch algorithm using training data. Next, the log-likelihood is calculated for each training datum using the Viterbi algorithm, in which the log-likelihood is divided by the word length as described above. The log-likelihoods of each datum are averaged in the word class. The averaged log-likelihood is used as the fitness of the HMM structure.

We set the number of states to be eight because, in our preliminary experiments, the HMM structure with eight states achieved the best score for spoken word recognition. We set 30 for the number of individuals in a generation. The GA procedure is independently performed for each word class.

In the initial condition, one of the individuals is set as a simple L-R HMM structure in which $a_{i,j} = a_{i,j+1} = 1$; others $= 0$, because the simple L-R structure achieved the highest score in our preliminary recognition experiments. The other 29 individual genotype strings are randomly generated.

For the selection, we adopt two strategies and combine them. One is the fitness-ordered strategy in which 29 candidate individuals for the next generation are selected in the following probability:

$$P_s \propto N - i, \quad (2)$$

where N is the population of a generation and i is the order of fitness. The other strategy is the elite-preservation strategy, in which an individual with the highest fitness always survives to be an individual of the next generation. Before the crossover, the candidates are randomly selected

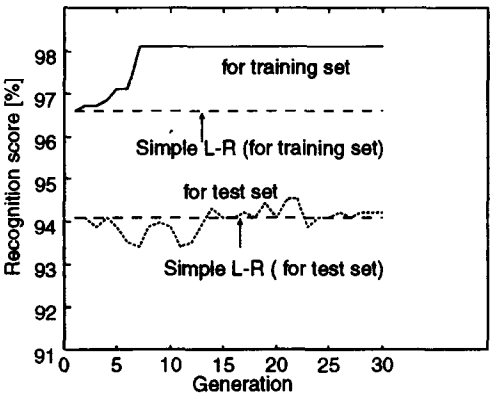

Figure 4: Only one HMM structure is used for all word classes.

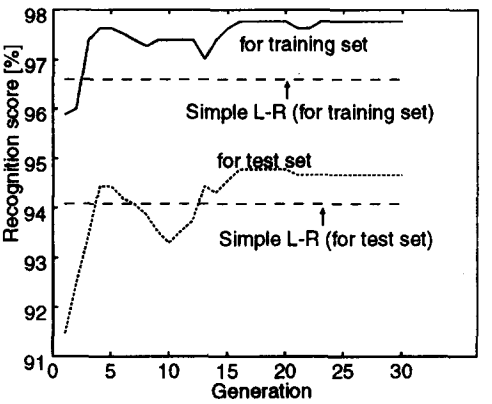

Figure 5: Proposed method.

and paired. Then the crossover operation is done for each pair. The crossover occurs in the probability 0.6 at one point in a genotype string, and two strings are generated. For the mutation, each bit of the string is inverted in the probability 0.03.

After the GA operations are repeated 30 times (or generations), the GA procedure is terminated.

4. RECOGNITION EXPERIMENT

In order to evaluate the proposed method, we performed recognition experiments. The speech data used in our recognition test are English numeral words from the database TIDIGITS[6]. For training, 11 numeral words "one" to "nine", "zero" and "oh" were uttered twice by 18 American males and 20 American females. We used 11 four-digit numerals uttered once by 20 males and 20 females including above mentioned speakers for code-book generation. In an open test, we used the same vocabulary of the above numeral words this time uttered by another group of 20 males and 20 females.

The speech sampling rate is 10kHz, and overlapping sections of 25.6ms of speech weighted by the Blackman window are analyzed every 10ms to give FFT power spectra. The power spectra are transformed to FMSs[7], which are the Fourier transforms of Mel Sone spectra whose frequency-axes are warped to be the mel scale and magnitude-axes are warped to be the sone scale. Three dimensional vectors, whose components are second to fourth components of the FMS, are used as the feature vectors. For code-book generation, we use the clustering algorithm[8] in which the FMS-space is repeatedly divided into two sub-spaces, obtaining cluster centers which minimize the estimation error at each sub-space. The code-book size is 64.

In order to compare to the proposed method, we performed two recognition tests. One is a conventional method without the GA using the simple L-R structure. Resulting recognition scores were 96.6% for the training data set and 94.1% for the test data set. These recognition scores are cited in the following graphs.

The other recognition test is one in which only one HMM structure is used for all word class. The GA is applied, however, the fitnesses are evaluated for the same HMM structure at each word class using the training set of each word class. The fitnesses are averaged over the classes, and the result is used as the fitness of the structure. Results of the recognition tests are shown in Figure 4, from which we can see that, for the training set, the recognition score becomes higher as the generation proceeds. For the test set, however, the recognition scores are around that of the conventional method.

In the proposed method, we set one of the initial individuals to be the simple L-R structure and monitor the recognition score whether it becomes higher or not than that of the simple L-R structure. Because we adopt the elite preservation strategy, we can certainly get better structures than the simple L-R structure whenever they exist. Result of the recognition test using the proposed method is shown in Figure 5. From this figure, we can see that the recognition score becomes higher as the generation proceeds. This is true not only for the training set but also for the test set. This shows that the selected structures are really effective. We performed the experiments three times and the same results were obtained. Consequently, it is shown that the genetic algorithm is effective for spoken word recognition.

5. DISCUSSION

The major features of the proposed method are (1) one of the initial individuals is the simple L-R structure, (2) the elite-preservation strategy is adopted, (3) the fitness-ordered strategy represented by the above mentioned expression (2) is adopted. We discuss here the effect of these features according to the recognition experiments.

Figure 6 shows the result of the recognition test in which the simple L-R structure is not set in the initial individuals. From this figure, we can see that the recognition score for the training set becomes around that of the simple L-R structure, and the recognition score for the test set improves slowly. This shows that the optimal structure is near to the simple L-R structure.

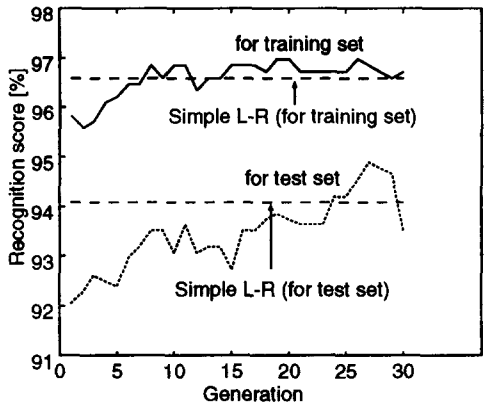

Figure 6: No simple L-R structure in the initial generation.

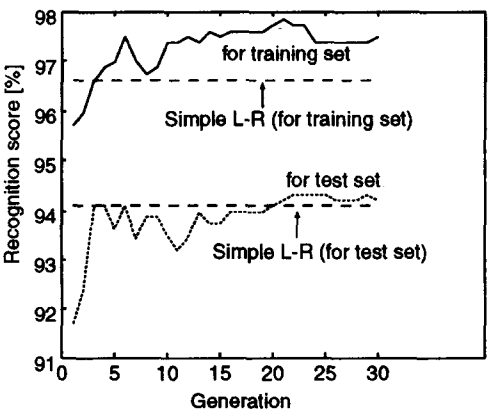

Figure 7: Without the elite-preservation strategy.

Figure 7 shows the result of the recognition test in which the elite-preservation strategy is not used. Comparing this figure to Figure 5 and Figure 6, we can see that the recognition score for the test set improves faster than that of Figure 6, but does not become much higher than that of the simple L-R structure. This result shows the effectiveness of the elite-preservation strategy of the proposed method.

Figure 8 shows the result of the recognition test in which the candidate of next generation is selected in the probability

$$P_s \propto 1/i, \tag{3}$$

where i is the order of fitness. From this figure, we can see that the recognition score for the test set becomes higher than that of the simple L-R structure in the early generations and declines to be around that of the simple L-R structure. This shows that the searching procedure falls into the structures with locally maximal recognition scores. Because it is not true in the proposed method, we see that the above mentioned expression (2) is effective for a wide scope search.

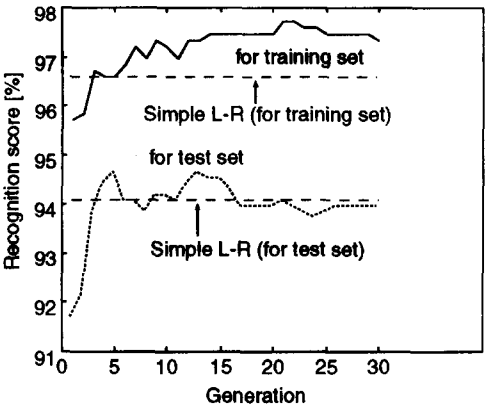

Figure 8: The other fitness-ordered strategy.

6. CONCLUSION

We applied the genetic algorithm to select the optimal HMM structure for isolated word recognition. Major features of this method are to use the log-likelihood per frame, to search around a simple left-right structure, to adopt an elite preservation strategy, and to adopt the fitness represented by the expression (2). We performed recognition experiments showing that the GA is effective for automatic speech recognition.

REFERENCES

[1] Rabinar, L. R. : "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", Proceedings of the IEEE, 77, 2, pp. 257-286 (Feb. 1989).

[2] Takami, J. and Sagayama, S. : "Automatic Generation of Hidden Markov Networks by a Successive State Splitting algorithm", (in Japanese) Trans. IEICE Japan, J76-D-II, 10, pp. 2155-2164 (Oct. 1993).

[3] Goldberg, D. E.: "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley Publishing Co., Inc., Reading, Massachusetts (1989).

[4] Takara, T. and Hirayasu A.: "Speech Recognition Using the Model Structure Determined by the Genetic Algorithm", Proc. of the Second World Congress of Nonlinear Analysts, to appear (Jul. 1996).

[5] Yada, T., Ishikawa, M., Tanaka, H. and Asai,K.: "Extraction of Hidden Markov Model Representations of Signal Patterns in DNA Sequences", Proc. of the First Pacific Symposium on Biocomputing, pp. 686-696 (1996).

[6] NIST: "TIDIGITS CD-ROM Set", NIST (Feb. 1991).

[7] Takara, T. and Imai S.: "Isolated Word Recognition Using DP-Matching and Maharanobis' Distance", (in Japanese) Trans. IECE Japan, J66-A, 1, pp. 64-70 (Jan. 1983).

[8] Nakagawa, S.: "Speech Recognition Using Probability Model", (in Japanese) pp. 18-26, Corona Co., Tokyo (1988).