Optimization of HMM by The Tabu Search Algorithm

Xiao-dan Mei [*], Sheng-he Sun [*], Jeng-shyang Pan [**] and Tsong-Yi Chen [**]

[*] Dept. of Automatic Test, Measurement and Control, Harbin Institute of Technology, China

[**] Dept. of Electronic Engineering, National Kaohsiung University of Applied Sciences, Taiwan

[**] jspan@cc.kuas.edu.tw , chentso@cc.kuas.edu.tw

Abstract

In this paper, the tabu search algorithm is employed to train Hidden Markov Model (HMM) to search out the optimal parameter structure of HMM for automatic speech recognition. The proposed TS-HMM training provided a mechanism that allows the searching process to escape from the local optimum and obtain a near global optimum. Experimental results show that the TS-HMM training has a higher probability to find the optimal model parameters than the traditional algorithms.

Keywords: tabu search, Hidden Markov Model, speech recognition, global optimum

1. INTRODUCTION

HMM is a highly robust statistical method and widely used for automatic speech recognition. It is a powerful algorithm used to estimate the model parameters and can achieve high performance [1,2,3]. Once a structure of the model is given, the model parameters are obtained automatically by feeding training data. The HMM model parameters take an important role in a HMM based speech recognizer because they can characterize the behavior of the speech segments and affect the system recognition accuracy directly.

Many heuristic algorithms are developed to optimize the model parameters in order to describe the trained observation sequences better, such as the forward-backward method [4] and the gradient method [5]. However, all these methods start from an initial guess and at last converge to a local optimum in practice. Few methods can escape from the local optimum to obtain the global optimum.

The tabu search algorithm [6] is the generalized heuristic global search technique with short-time memory, and suitable for solving many nonlinear optimization problems. The basic idea of the tabu search approach is to explore the search space of all feasible solutions by a sequence of moves. The spirit of this method is embedded in its short-term memory process. The elements of the move from the current solution to its selected neighbor are partially or completely recorded in the tabu list for forbidding the reversal of the replacement in future iterations. The search would cycle between the first encountered local optimum and its neighbor without this assurance.

In this paper, the tabu search algorithm is utilized to HMM training to search out the optimal structure of HMM for automatic speech recognition. The proposed TS-HMM training provided a mechanism that allows the searching process to escape from the local optimum and to obtain a near global optimum.

In Section 2 of this paper, the definition of HMM is given, then the tabu search algorithm is described in Section 3. The TS-HMM training algorithm is presented in Section 4. Simulation results are shown in Section 5 and conclusions are given in Section 6.

2. HIDDEN MARKOV MODEL

HMM is a probability model used to represent the statistic property of the stochastic process and is characterized by the model parameters. The stochastic process in speech recognition is the finite-length stochastic sequences called observation symbol, denoted by $\boldsymbol{O} = \boldsymbol{o}_1\boldsymbol{o}_2\cdots\boldsymbol{o}_M$, where M is the dimension of the observation symbol. One HMM with N states (S_1, S_2, $\cdots$, S_N) can be characterized by the parameter set $\boldsymbol{\lambda} = \{\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}\}$, where

(1) $\boldsymbol{\pi} = [\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \cdots, \boldsymbol{\pi}_N]$ is the initial distribution. It is used to describe the probability distribution of the observation symbol in the initial moment when $t = 1$, namely

$$\boldsymbol{\pi} = P(q_1 = S_i) \qquad i = 1, 2, \cdots, N \tag{1}$$

it must satisfy $$\sum_{i=1}^{N} \boldsymbol{\pi}_i = 1. \tag{2}$$

(2) $\boldsymbol{A}=\{a_{ij} \mid i, j=1, 2, \cdots, N\}$ is the transition probability distribution matrix. Its element at row i, column j is the probability a_{ij} of transition from current state i to next state j, namely

$$a_{ij}= P\,(q_t= S_j \mid q_{t-1}= S_i) \tag{3}$$

it must satisfy the following condition:

$$\sum_{j=1}^{N} a_{ij} = 1\ . \qquad (4)$$

(3) $\boldsymbol{B}=\{b_{ik} \mid i=1, 2, \cdots, N, k=1, 2, \cdots, M\}$ is the observation symbol probability distribution matrix in the discrete HMM. Its element at row i, column k is the probability b_{ik} of observation symbol with index k emitted by current state i and must satisfy the following condition:

$$\sum_{k=1}^{M} b_{ik} = 1\ . \qquad (5)$$

As above-mentioned, HMM is used to approximate the probability of each observation symbol existing in the current state. When $\boldsymbol{\pi}$, $\boldsymbol{A}$, $\boldsymbol{B}$ are given, the probability $P\ (\boldsymbol{O}|\boldsymbol{\lambda}\ )$ of the HMM system generating one random observation symbol can be calculated. Three essential problems of HMM must be solved, they are:

(i) how to effectively calculate the probability $P\ (\boldsymbol{O}|\boldsymbol{\lambda}\ )$;

(ii) how to select the optimal state sequence when the model $\boldsymbol{\lambda}$ is given;

(iii) how to adjust the model parameter $\boldsymbol{\lambda}$ to make the probability $P\ (\boldsymbol{O}|\boldsymbol{\lambda}\ )$ higher.

People often employ the Forward algorithm to solve the first problem and the Viterbi algorithm to solve the second one. For the third problem, people use Gradient algorithm. This paper aims at solving the last problem, we use the tabu search algorithm to search the optimal model parameters $\boldsymbol{\lambda}$.

3. THE TABU SEARCH ALGORITHM

The tabu search algorithm, which was proposed by Glover [6], is a generalized heuristic global search technique with short-time memory. Its basic idea is to explore the search space of all feasible solutions by a sequence of moves and to forbid some search directions at the present iteration in order to avoid cycling and jump off local optima. The elements of a move from the current solution to its selected neighbor are partially or completely recorded in the tabu list for the purpose of forbidding the reversal of the replacement in a number of future iterations.

The tabu search approach begins with test solutions generated randomly and their corresponding objective function values are computed. If the best of these solutions is not tabu or if it is tabu but satisfies the aspiration criterion, then select this solution to be the new current solution to generate test solutions for next iteration. It is called aspiration criterion if the test solution is a tabu solution but the objective value is better than the best value of all iterations. The tabu search algorithm is given as follows:

```
Tabu Search Algorithm ()
{
    generate the initial solutions;
    calculate the current solution and the best solution;
    while termination criterion not reached
    {
        generate the test solutions in the neighborhood of the current solutions;
        calculate the corresponding objective values;
        update the current solution and the best solution;
        update the tabu list;
    }
}
```

4. THE TS-HMM ALGORITHM

In this paper, the configuration of HMM is a five states left-right model and the speech feature vectors are vector quantized into the codebook with the size of 256. So $\boldsymbol{A}$ is a 5-by-5 matrix and $\boldsymbol{B}$ is a 5-by-256 matrix. As shown in Fig. 1, this model can represent speech signal whose properties change over time in a successive manner.

Due to the configuration of the model, some transitions between states do not exist so that the corresponding elements in matrix $\boldsymbol{A}$ are constantly zero and these elements will not be encoded when performing search.

The optimal model parameters searching problem must be mapped to the tabu search algorithm before it can be used. The mapping procedure is described as follow:

In TS-HMM training, the model is encoded into a string of real numbers between 0 and 1, and of course they satisfy the equation (4) and (5). As shown in Fig. 2, this string is composed of two parts: ***SA*** and ***SB***. These two parts are composed of the rows of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ respectively.

A solution of this algorithm is defined as $\boldsymbol{s}_l$ consisting of a set of real numbers like the one shown in Fig. 2. The probability $\boldsymbol{p}_n(\boldsymbol{O}|\boldsymbol{\lambda}_n)$ of the HMM solution λ_n which generates the training observation sequences $\boldsymbol{O} = \boldsymbol{o}_1\boldsymbol{o}_2\cdots\boldsymbol{o}_M$ must be calculated as the objective function value.

The initial test solutions are generated randomly. After the first iteration, the test solutions are generated from the best solution of current iteration by swapping two indicies randomly. The tabu list memory stores the swapped indices only. It is a tabu condition if the swapped indicies to generate the new test solution from the best solution of current iteration are the same as any records in the tabu list memory.

Let $\boldsymbol{\theta}_t = \{\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \cdots, \boldsymbol{\lambda}_{N_s}\}$ to be the set of the test solutions, let $\boldsymbol{\lambda}_c = \{\boldsymbol{\lambda}_c(1), \boldsymbol{\lambda}_c(2), \cdots, \boldsymbol{\lambda}_c(N)\}$ and $\boldsymbol{\lambda}_b = \{\boldsymbol{\lambda}_b(1), \boldsymbol{\lambda}_b(2), \cdots, \boldsymbol{\lambda}_b(N)\}$ be the best solution of current iteration and the best solution of all iterations respectively, let $V_t=\{v_1, v_2, \ldots, v_{Ns}\}$, v_c and v_b denote the set of objective function values for test solutions, the objective function value for the best solution of current iteration and the objective function value for the best solution of all iterations, respectively, where v_l is the objective function value for solution $\boldsymbol{\lambda}_l$, $1 \le l \le N_s$. The algorithm is given as follows:

Step 0. Set the tabu list size T_s, the number of test solutions N_s and the optimum number of iterations l_m. Set the iteration counter $i = 1$ and insertion point of the tabu list $t_l = 1$. Generate N_s solutions $\boldsymbol{\theta}_t = \{\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \cdots, \boldsymbol{\lambda}_{N_s}\}$ randomly, calculate the corresponding objective values $V_t=\{v_1, v_2, \ldots, v_{Ns}\}$ and find out the current best solution $\boldsymbol{\lambda}_c = \boldsymbol{\lambda}_j$, $j = \arg\max_l (v_l)$, $1 \le l \le N_s$. Set $\boldsymbol{\lambda}_b = \boldsymbol{\lambda}_c$ and $v_b = v_c$.

Step 1. Copy the current best path $\boldsymbol{\lambda}_c$ to each test solution $\boldsymbol{\lambda}_l$, $1 \le l \le N_s$. For each test solution $\boldsymbol{\lambda}_l$, $1 \le l \le N_s$, generate two random integers r_1 and r_2, $1 \le r_1 \le N$, $1 \le r_2 \le N$, $r_1 \ne r_2$. Generate the new test solutions by swapping $\boldsymbol{\lambda}_l(r_1)$ and $\boldsymbol{\lambda}_2(r_2)$. Calculate the correspanding objective values $v_1, v_2, \cdots, v_{N_s}$ for the new test solutions.

Step 2. Sort $v_1, v_2, \cdots, v_{N_s}$ in increasing order. From the best test solution to the worst test solution, if the test solution is a non-tabu solution or it is a tabu solution but its objective value is larger than the best value of all iterations v_b (aspiration level), then choose this solution as the current best solution $\boldsymbol{\lambda}_c$ and choose its objective value as the current best objective value v_c,go to step 3; otherwise,try the next test solution. If all test solutions are tabu solutions, then go to step 1.

Step 3. If $v_c < v_b$, set $\boldsymbol{\lambda}_c = \boldsymbol{\lambda}_b$ and $v_c = v_b$. Insert the swapped indices of the current best solution $\boldsymbol{\lambda}_c$ into the tabu list. Set the inserting point of the tabu list $t_l = t_l + 1$. If $t_l > T_s$, set $t_l = 1$. If $i < l_m$, set $i = i + 1$ and go to step 1; otherwise, record the best path index and terminate the algorithm.

5. SIMULATIONS

10 experiments are conducted to validate the algorithm proposed in this paper. We recorded each word's pronunciation 10 times. Then we have 100 training observation sequences. For each word, we used the tabu search algorithm and the forward-backward algorithm to train the HMM respectively, and then we can obtain two sets of HMM model parameters and compare them. In this paper, the length of the tabu list $T_s = 20$, the threshold of the probability $P_{th} = 0.17\%$, the number of the iteration I_m =800, the number of the solutions in each iterationN_S=20. The initial model parameters are created randomly and are normalized to satisfy the equation (4) and (5).

In each experiment, the HMM training using the forward-backward algorithm will be terminated when the increase of the average log probability less than 0.00001 and TS-HMM training will be terminated after 800 iterations.

In this paper, we compared the HMMs trained by the tabu search algorithm and the forward-backward algorithm respectively. Simulation results are shown in Table 1. They are made up of two parts: P_s and P_d. P_s denotes the average log probability of the HMM generated by the 10 training observation sequences of this HMM and P_d denotes the average log probability of the HMM generated by the other 90 training observation sequences of the other HMMs.

As shown in Table 1, the HMMs trained by the tabu search algorithm that have higher average log probabilities than the HMMs trained by the forward-backward algorithm except experiment #6. It means the HMMs trained by the tabu search algorithm can better describe and recognize the training observation sequences. The experiment #6 is not satisfying because the better optimum is not encountered during searching, thus the whole search procedure is not globally optimal.

6. CONCLUSIONS

This paper proposes the TS-HMM training method. The tabu search algorithm is employed to repair the HMM model parameters λ and make $P_n(O|\lambda\ )$ highest. The simulation results indicated that TS-HMM training has a higher probability in finding the global optimal parameters with better performance than the forward-backward algorithm. Besides, parallel implementation of TS algorithm can be employed to reduce searching time such that its searching time can compare with other heuristic algorithms.

REFERENCE

1. L. R. Rabiner, "A tutorial on Hidden Markov Models and Selected applications in speech recognition," *Proceeding of IEEE*, , Vol. 77, No. 2, 1989, pp. 257~285.
2. Joseph Picone, "Continuous speech recognition using Hidden Markov Models," *IEEE ASSP Mag.*, Vol. 7, No. 7, 1990, pp. 26~41.
3. D. Burshtein, "Robust parametric modeling of duration in Hidden Markov Models," *IEEE Trans. on Speech & Audio Processing*, , Vol. 4, No. 3, 1996, pp. 240~242.
4. F. Jelinek, "Continuous speech recognition by statistical methods," *Proceeding of IEEE*, Vol. 64, No. 4, 1976, pp. 532~556.
5. S. E. Levinson, L. R. Rabiner and M. M. Sondhi, "An introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition," *The Bell System Technical Journal*, April 1983, pp. 1035~1074.
6. F. Glover and M. Laguna, "Tabu search," *Kluwer Academic Publishers*, 1997.

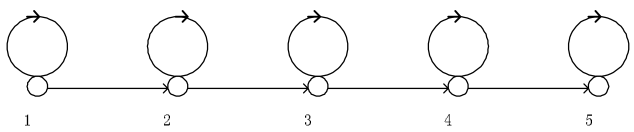

Fig. 1. A five states left-right model

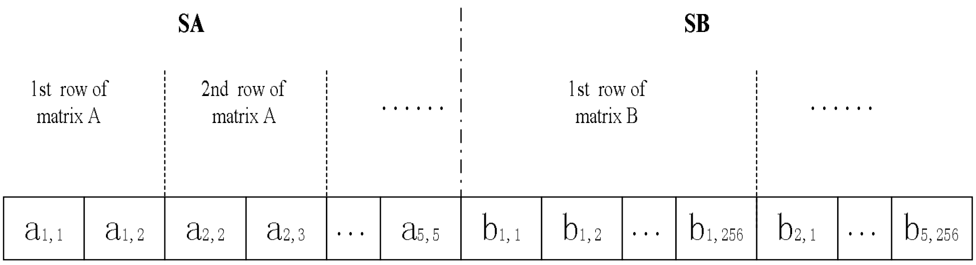

Fig. 2. The string representation of HMM

Table 1 The comparison of average log probability obtained with two algorithms

Experiment	TS		Forward-Backward	
	P_s	P_d	P_s	P_d
#1	-3.3463	-9.0765	-4.2946	-8.9249
#2	-4.8036	-9.4363	-4.8116	-8.4035
#3	-4.6056	-8.4663	-5.6599	-8.3756
#4	-3.5379	-8.3139	-4.3562	-7.9967
#5	-4.6579	-9.9391	-5.1033	-7.6877
#6	-4.5324	-9.3661	-4.3394	-8.6031
#7	-3.2752	-9.3218	-4.7167	-8.4162
#8	-3.6225	-9.3123	-4.3607	-8.2275
#9	-3.8032	-9.6469	-4.5107	-9.2521
#10	-4.3190	-8.2152	-4.4864	-7.8755