English Speech Recognition System Based on HMM in Matlab

YANG Xiaocui[1], SUN Lihua[2]

College of Information Engineering, Nanchang University, Nanchang, China

e-mail: yxczcl@163.com, slh52@163.com

Abstract: Through the study of medium-vocabulary speaker independent continuous English speech recognition algorithm, 100(910 words) common tourism English sentences are simulated using Voice-box in MATLAB toolbox. Since it is speaker independent speech recognition, the experiments were conducted through five male and five female for HMM (Hidden Markov Models) and the English speech recognition tests, the correct rate of the speech recognition tests is more then 96%. The innovation point is that speech recognition is applied in Tourism English Training, the training is for speaker independent, the test result of speech recognition displays in scores, and this can let the testers to understand their pronunciation accuracy.

Keywords: english speech recognition; MFCC (mel-frequency cepstral coefficients); CDHMM (continuous density hidden markov models)

1 Introduction

English is very important and useful as an international language. More and more people travel abroad with the development of China's economy, but some people do not know English, which is very inconvenient and leads to barriers in communication with foreigners, thus, some people hope to learn some common English through English training software. This makes specialized English recognition software become popular. Based on this situation, the paper makes intensive study of medium-vocabulary speaker independent continuous English speech recognition algorithm and then some common Tourism English sentences are simulated in matlab.This is very important to the development of portable useful embedded English recognition software.

2 The Flow Chart of Medium-Vocabulary English Speech Recognition

Figure 1 displays the flow chart of speech recognition, which has three parts: feature extraction, training and recognition [1]. The observation vectors in feature extraction are trained firstly, and then they are optimized in training, finally, the optimized vectors are stored in recognition model storeroom. In speech recognition, we also need to make feature extraction and create observation vectors, and then the observation vectors will be compared with the vectors in model storeroom and the corresponding result of recognition will be found. For different type of speech recognition, different means are adopted to improve the correct rate of recognition.

2.1 Features Extraction

After digitalization and preprocessing, we should make endpoint detection and features extraction of speech

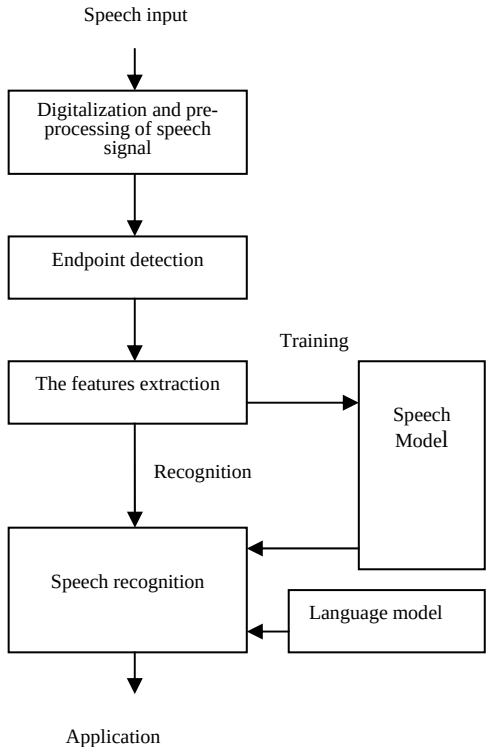

Figure 1. The flow chart of speech recognition

signal. Digitalization includes the voltage amplification of speech signal, auto gain control, anti-aliasing filter, sampling, analog to digital conversion and coding. Preprocessing includes pre-emphasis, windowing, frame segmentation, etc[2]. The goal of features analysis is to

978-1-935068-06-8

extract the features of speech and to make the distance in cluster as short as possible while the distance between clusters as long as possible and the choice of features affects the accuracy of speech recognition. There are many features such as cepstrum, linear prediction coefficient, frequency spectrum, average energy and MFCC coefficients.

In this paper, we chose MFCC which is different to common practical frequency cepstral coefficients. MFCC bases on the acoustical characteristic and accords with people's acoustical characteristic. MFCC is built on Flourier frequency spectrum analysis, its core idea is to use the perception characteristics of ears, set up band pass filter with triangle or sine characteristic in the frequency range of speech, calculate the signal energy of according filters, apply logarithm to the output of filters, lastly calculate corresponding MFCC through DCT (Discrete Cosine Transformation)[3].

2.2 Training and Recognition

Speech recognition needs to build model storeroom that is the training of speech model. For medium-vocabulary speaker independent continuous English speech recognition system, we adopt continuous HMM model. Continuous HMM can process speech data directly, However in order to have good recognition performance, we must mix many probability functions to gain the probability density function of a state's corresponding observation value. When we use HMM as the recognition model, the output of feature vectors and the matching of speech input and speech model will waste a lot of time and space. In order to reduce the storage and complexity of calculation, we can analyze speech or model vector quantization and use the central value representing speech feature for match [4]. Because the paper bases on personal computer, the reduction of calculation is more important than the reduction of storage. In HMM training, we use Baum-Welch algorithm.

In the speech recognition process, input speech should also pass digitizing, pre-processing, endpoint detection and features extraction, then we will gain the observation sequence, next we can find the best state sequence through Viterbi algorithm, lastly we can have the word through check storeroom according the best state sequence.

2.3 Syntactic Analysis and Word Prediction

In continuous speech recognition, speech recognition processing and language syntax analysis are always combined following the priority order. Firstly acoustics model matches with input speech, and then we will get a batch of words, next we will find the best words sequence according with the restriction of syntax through language model of input speech. But if the speech processing and language processing are not isochronous, computation will become complicated and recognition precision will be affected. The improved method is to let language processing combine with speech processing frame synchronously.

There are several sub-word units that can be used to model speech; they are phone-like units, syllable-like units, dyad or demisyllable-like units and acoustic units [5]. Because each of the above subword unite sets can represent any word in the English language, the problems in the choice of subword unit sets are the context sensitivity and the ease of training the unit from fluent speech. The number of phone-like units is small; typically there are about 50 phone-like units for English. The set of phone-like units is extremely context sensitive because each unit is potentially affected by its predecessors and its followers. For medium-vocabulary speaker independent continuous English speech recognition, we choose phone-like units for simplicity.

3 HMM Algorithm

Hidden Markov models (Hidden Markov Model) is a major development in the speech recognition field in1980s. On the one hand, the hidden states correspond to the relatively stable pronunciation units of acoustic layer, and the state transition and State presence descript changes in pronunciation. On the other hand, it introduces the probability statistical model and calculates the output probability of HMM model of speech characteristic parameters using probability density function, then find recognition results by searching best state sequence and maximum posteriori probability criterion. It uses training methods to make the bottom acoustic model and upper language model merge into a unified speech recognition search algorithm. At present, speech recognition algorithm based on the HMM model is the main stream. The paper use Continuous density Hidden Markov Model (CDHMM) to every order [2-5].

3.1 Continuous Density Hidden Markov Model

Each state observation probability density function of CDHMM model is described by the linear combination of n consecutive Gaussian probability density function. Each consecutive Gaussian probability density function has mean vector and covariance matrix. So HMM model is described by $\lambda(A,B,\pi)$, its output probability density function is:

$$B=\{b_j(O)\}; \tag{1}$$

$$b_j(O)=\sum_{i=1}^{M} c_{jk} N(O,\mu_{jk},U_{jk}), \quad 1\le j\le N \tag{2}$$

Here $O=(O_1O_2\cdots O_T)$ is the given observation se-

quence, M is the number of distinct observation symbols per state, c_{jk} is the k-relation mix Gaussian function under j state, N on behalf of normal Gaussian probability density function, μ_{jk} and U_{jk} represent respectively for the k-relation mix Gaussian mean vector and covariance matrix under j state. Weight coefficients c_{jk} meet the following conditions:

$$\sum_{i=1}^{M} c_{jk} = 1, 1 \le j \le N \tag{3}$$

3.2 The Solution to There Basic Problems for Hmms

3.2.1 Problem One

Given the observation sequence $O = (o_1 o_2 \cdots o_T)$, and a model $\lambda = (A, B, \pi)$, how do we efficiently compute $p(O/\lambda)$, the probability of observation sequence, given the model? We usually use the forward procedure and backward procedure. The forward variable defined as

$$\alpha_t(i) = p(o_1 o_2 \cdots o_t, q_t = i \mid \lambda) \tag{4}$$

That is probability of the partial observation sequence, $o_1 o_2 \ldots o_t$, (until time t) and state i at time t, given the model λ. We can solve for $\alpha_t(i)$ as follows:

1) Initialization

$$\alpha_1(i) = \pi_i b_i(o_1), 1 \le i \le N \tag{5}$$

2) Induction

$$\alpha_{t+1}(j) = [\sum_{i=1}^{N} \alpha_t(i) a_{ij}] b_j(o_{t+1}) \quad 1 \le t \le T-1, 1 \le j \le N \tag{6}$$

3) Termination

$$p(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{7}$$

Similarly, we can consider a backward variable

$$\beta_t(i) = p(o_{t+1} o_{t+2} \cdots o_{t+3} \mid q_t = i, \lambda) \tag{8}$$

That is the probability of the partial observation sequence from $t+1$ to end, given state i at time t and the model λ. We also can solve for $\beta_i(i)$ as follows:

1) Initialization

$$\beta_T(i) = 1, 1 \le i \le N \tag{9}$$

2) Induction

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j), \quad t = T-1, T-2, \ldots, 1, 1 \le i \le N \tag{10}$$

We can compute the whole output probability is

$$p(o \mid \lambda) = \sum_{i=1}^{N} \alpha_t(i) \beta_t(i) = \sum_{i=1}^{N} \alpha_T(i), \quad 1 \le t \le T-1 \tag{11}$$

Another form is

$$p(o \mid \lambda) = \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j), \quad 1 \le t \le T-1 \tag{12}$$

3.2.2 Problem Two

Given the observation sequence $O = (o_1 o_2 \ldots o_T)$, and a model λ, how do we find a corresponding state sequence $q = (q_1 q_2 \ldots q_T)$ that is the optimal in some sense? To solve this problem we usually use Viterbi Algorithm. To find the single best state sequence $q = (q_1 q_2 \ldots q_T)$, for the given observation sequence $O = (o_1 o_2 \ldots o_T)$, we define the quantity

$$\delta_t(i) = \max_{q_1, q_2, \ldots, q_{t-1}} p[q_1 q_2 \ldots q_{t-1}, q_t = i, o_1 o_2 \ldots o_t \mid \lambda] \tag{13}$$

$\delta_t(i)$ is the best score (highest probability) along a single path, at time t, which accounts for the first t observations and ends in state i. By induction we have

$$\delta_{t+1}(j) = [\max_i \delta_t(i) a_{ij}] \cdot b_j(o_{t+1}) \tag{14}$$

To actually retrieve the state sequence, we need to keep track of the argument that maximized for each t and j. We do this via the array $\psi_t(j)$. The complete procedure for finding the best sequence can be stated as follows:

1) Initialization

$$\delta_1(i) = \pi_i b_i(o_1), 1 \le i \le N \tag{15}$$

$$\psi_1(i) = 0 \tag{16}$$

2) Recursion

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] b_j(o_t), \quad 2 \le t \le T, 1 \le j \le N \tag{17}$$

$$\psi_t(j) = \arg\max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}], \quad 2 \le t \le T, 1 \le j \le N \tag{18}$$

978-1-935068-06-8 © 2009 SciRes.

3) Termination

$$p^* = \max_{1\le i\le N}[\delta_T(i)] \tag{19}$$

$$q_t^* = \arg\max_{1\le i\le N}[\delta_T(i)] \tag{20}$$

4) Path backtracking

$$q_t^* = \psi_{t+1}(q_{t+1}^*), t = T-1, T-2, ..., 1 \tag{21}$$

3.2.3 Problem Three

How do we adjust the model parameters $\lambda = (A, B, \pi)$ to maximize $p(o \mid \lambda)$?

To solve the parameter estimation, we first define $\xi_t(i, j)$ and $\gamma_t(i)$, $\xi_t(i, j)$ is the probability of being in state i and t, and state j at time $t+1$. $\gamma_t(i)$ Is the probability of being in state i at time t, Giving

$$\xi_t(i, j) = p(q_t = i, q_{t+1} = j \mid O, \lambda) \tag{22}$$

$$\gamma_t(i) = p(q_t = S_i \mid O, \lambda) \tag{23}$$

Then we get

$$\xi_t(i, j) = \frac{\alpha_t a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{p(O \mid \lambda)} \tag{24}$$

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i, j) \tag{25}$$

Using the above formulas, we can get a set of reasonable reestimation formulas for π, A, and B.

$$\overline{\pi_i} = \gamma_i(t) \tag{26}$$

$$\overline{a_{ij}} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \tag{27}$$

$$\overline{b_{ij}} = \frac{\sum_{t=1, O_t=0}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)} \tag{28}$$

$\overline{\lambda} = (\overline{A}, \overline{B}, \overline{\pi})$ is reestimation parameter, and $p(O \mid \overline{\lambda}) \ge p(O \mid \lambda)$.

4 The Simulation on Matlab and The Analysis of Results

Matlab supports NeXT/SUN SPARC station sound files(suffix is .au), Microsoft Wave sound files (suffix is .wav), windows-compatible sound equipments, sound recording and broadcasting, as well as the audio signal of linear law and the audio signal of mu law. MATLAB can read, write, get sound information and etc. At the same time as a high-level language integrated development environment, Matlab can create GUI (graphical user interface) programs [6]. GUI programs are components realizing the communication between interface display and user, the user can click interactive components of mouse and keyboard to achieve a particular function, and then the output of MATLAB will display in the corresponding results area [7]. Fig. 2 shows the GUI for English speech recognition used in experimental process. It is very convenient for speech recognition.

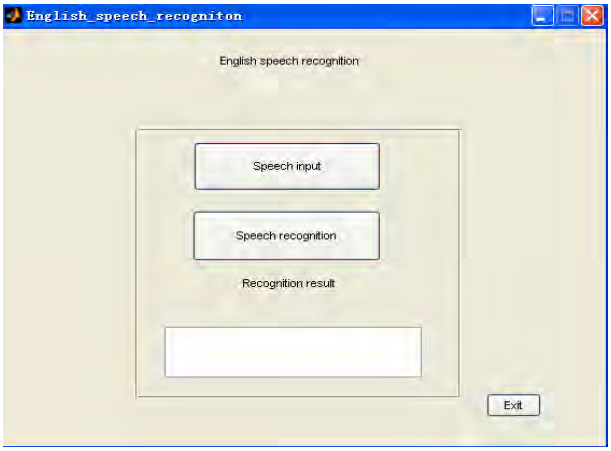

Figure 2. The GUI of English speech recognition

Based on the familiarity of medium-vocabulary speaker independent continuous English speech recognition algorithm, 100 common traveling English sentences including 910 words are simulated in matlab environment based on pc. To speaker independent speech recognition, the speech model of speaker independent needs a large number of speech data for training. So we choose 10 persons for test. They are tested under quiet and noisy condition using good microphone. We recorded and trained 10 random sentences of each person. The correct recognition rate of noisy condition is 95%, the correct recognition rate of quiet condition is 97%.

5 Acknowledgements

This paper can be published smoothly with a lot of people especially my tutor who are inseparable for their selfless help, they encouraged me and helped me to analyze my problems.

References

[1] Zhao Li. Speech recognition processing M], China Machine Press, 2008, P212-221 (Ch).
[2] Yao Jing, Wang Guoliang, Liu Jia. The algorithm of embedded English commanding words speech recognition [J]. Micro-computer Information, 2008.6-2, P4-8 (Ch).
[3] Zhang Zhen, Wang Huaqing. The improved algorithm of Mel-frequency Cepstral Coefficients in features extraction of speech signal [J]. Computer Engineering and Application. 2008, 22, P54-55 (Ch).
[4] Liang Wenbin, Zhang Fan, Cheng Jing, Zhao Xinkuan. The realization of embedded real-time music speech recognition [J]. Micro-computer Information, 2008.7-1, P252-253 (Ch).
[5] Lawrence Rabiner, Biing-Hwang Juang. Fundamentals of speech recognition [M].1999, P434-439.
[6] Long Yingdong, Liu Yuhong, Jing Lan, Qiao Weimin. The realization of speech recognition on matlab[J].Micro-computer Information, 2007, 12-1, P255-256 (Ch).
[7] He Qiang, He Ying. The expansion of programming on matlab[M]. Peking University Press, Beijing, 2002, P330-358 (Ch).