Improved Viterbi Algorithm in Continuous Speech Recognition

LIU Wei, HAN Weisheng
School of Electronic Information Engineering
Henan University of Science and Technology
Luoyang, China, 471003

***Abstract*—It is key technique to improve the recognition efficiency while not affecting the recognition accuracy in the speech recognition systems. The method combining Viterbi algorithm with Beam pruning technique is useful to compress the search space, which reduces the computational complexity. However, Viterbi-Beam algorithm is a kind of suboptimal algorithm. The selecting of the pruning threshold will affect the system recognition accuracy. In this paper, we propose an adaptive Viterbi-Beam search algorithm by analyzing the voice activity model of different stages. This method leads to an improvement on search efficiency of 35.77%, without the recognition accuracy reduction.**

Keywords-speech recognition; hidden Markov model; Viterbi-Beam algorithm; pruning threshold

I. INTRODUCTION

Speech recognition problem is to effectively search out the best sequence in the space consisting of observation sequence, language model and acoustic model. The search process is also known as decoding. Classical Viterbi algorithm can successfully solve the HMM (Hidden Markov Model) decoding problem. However, the exhaustive search through the network is intractable in large vocabulary continuous speech recognition (LVCSR) tasks[1]. Viterbi-Beam search algorithm, combining Viterbi algorithm with Beam technique, achieve the purpose of reducing calculation by introducing pruning.

Ideal pruning restricts the search space efficiency without degrading the recognition accuracy. The traditional Viterbi-Beam algorithm is good candidate in search algorithms. It restricts the search space to make the recognition feasible, and minimize the number of search errors by choosing the appropriate pruning threshold [2] .

In this paper, we firstly analysis the traditional Viterbi-Beam algorithm in speech recognition system, and then propose an improved adaptive search algorithm. The experiments and performance comparison between the two recognition algorithms in terms of efficiency and accuracy are also presented.

II. TRADITIONAL VITERBI-BEAM SEARCH ALGORITHM

The idea of the search pruning is to retain only the most promising path hypotheses as the starting points for the following path expansion. The relative optimum of the paths can be determined by their likelihood scores[3].

The highest path probability score at time t can be approximated as:

$$Q_{\max}(w,t) = \max_{w}\{Q(t, s=0; w)\} \quad (1)$$

$$Q_{\max}(s,t) = \max_{s}\{Q(t, s; w)\} \quad (2)$$

The pruning thresholds of word level and state level are fixed, denoting the beam width of which $b(w)$ and $b(s)$ respectively. The pruning criterion can be expressed as follows

$$Q(t, s=0; w) > Q_{\max}(w,t) - b(w) \quad (3)$$

$$Q(t, s; w) > Q_{\max}(s,t) - b(s) \quad (4)$$

Obviously, the path will be abandoned as long as it does not fulfill (3) and (4). Because the traditional Viterbi-Beam search algorithm uses a fixed pruning threshold, it is easier to realized. But when a path is very prominent, it may only keep this path while the real global optimal path is excluded, resulting in decreasing the system recognition accuracy. On the other hand, too many paths will be kept in some frames whose probability distribution are concentrated, resulting in decreasing the system recognition efficiency. The process of traditional Viterbi-Beam search algorithm as follows:

```
•Initialization:
  Active path
•Recursion:
  For t = 1 to T
      For each state i
          Time-alignment: calculate Q(t, s; w)
          Carry out state pruning
      End
      For each word
       Calculate Q(t-1, s=0; w)
      Carry out word pruning
      End
  End {observation vector sequence}
```

III. ADAPTIVE VITERBI-BEAM SEARCH ALGORITHM

Considering the deficiencies of traditional Viterbi-Beam search algorithm, it is necessary to adaptively change pruning thresholds during decoding process. Adaptive Viterbi-Beam search algorithm can dynamically adjusts the value of pruning threshold based on the output information sequence during the search course. At present, the dynamic threshold is adjusted normally based on the highest path score and the average score. We denote as $b_s(t)$ the pruning threshold state[4], which can be expressed as

$$b_s(t) = \frac{1}{N}\sum_{i=1}^{N}(s_{\max} - s_i) \quad (5)$$

Where $s_{\max}$ is the highest likelihood score of current paths, s_i represents the likelihood score of states i, and N is the total number of active states model. From equation (5), we can obtain that the state pruning threshold is the difference of the highest path score and the average score. Although experimental results show that the method can improve the recognize efficiency by 14%, there are two problems in search algorithm.

(1) If most of scores in active models are low, the average score of the path becomes too small, which leads to the difference $b_s(t)$ increases, resulting in decreasing the recognition efficiency.

(2) In the sure stage models, $b_s(t) \to 0$ in short time, resulting in no output.

Therefore, it is unreasonable to determine the pruning threshold only relying on the highest score and the average path score. Then, this article introduces a new adaptive Viterbi-Beam search program.

According to speech features, the Viterbi-Beam search process can be divided into three phases

(1) *Phase one* (first 50 ~ 60 frames): At the beginning of search ,It is difficult to decide the starting word just depending on the feature vectors of 100 ~ 200ms, all path scores change slightly, so we can use a smaller pruning threshold; With the search carrying on, the scores increases rapidly, we should expand the beam width accordingly.

(2) *Phase two* (next 150 to 200 frames): Although the scope change of likelihood scores in this phase become greater, local search is the main work , so we need to maintain the pruning threshold to a certain value .

(3) *Phase three* (rest of speech): In this stage, the scores of different paths vary greatly. Only few paths will be kept even with large pruning threshold.

Through the above analysis, we can see that the first phase is extremely important. If the pruning threshold of the first phase is too small, we may have a high search error, affecting the accuracy of system recognition. Inversely, if the pruning threshold in the first phase is too big, it would be unbeneficial to local search in the second phase, affecting the system recognition efficiency. Thus, we can come up with a new pruning strategy: increasing the pruning threshold in the first phase adaptively, and keeping the pruning threshold in the second phase and the third three. Accordingly, we propose an improved adaptive Viterbi-Beam search algorithm.

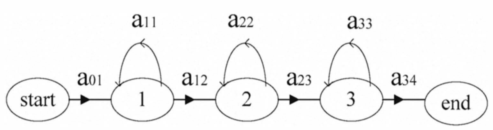

Figure 1. A typical Left-Right HMM model

Figure 1 is a typical Left-Right HMM model, including the start state and end state. A model without any pruning will surely produce a new model from the activities of the transfer path to another activity model after three of the feature vector, which means model feature vector in the presence of an active period is at least three. After three frames will transfer to other models, it results a new activity model. We know that in the first phase, the active model number is increasing. Therefore, if the number of model at this stage does not increase after three consecutive frames, we can determine the pruning threshold is too small and determine another active pruned model. We need to extend threshold adaptively. Defining $\theta_s(t)$ as increscent value of pruning threshold, defined as:

$$\theta_s(t) = \alpha(s_{\max} - (b_s(t) + s_{p-avg})),\ 1 < t < 50 \quad (6)$$

where $s_{\max}$ indicates the current highest score path, $b_s(t)$ is the state pruning threshold path at time t, s_{p-avg} is the average score of abandoned paths. α is a constant[5]. The realization process of optimized Viterbi-Beam search algorithm as follows

- Initialization:
  - Active path
  - $b_s(1) = 5$
- Recursion:

```
For t = 1 to T
    For each level i
        Time-alignment: calculate Q(t, s; w)
        Carry out State pruning
        If 1 < t < 50 and the number of active model
              don't increase
            Calculate θ_s(t)
            b_s(t) = b_s(t) + θ_s(t)
        End
    End
    For each word
        Calculate Q(t-1, s = 0; w)
        Carry out word pruning
    End
End {observation vector sequence}
```

IV. EXPERIMENT AND RESULTS

According to the adaptive Viterbi-Beam algorithm, the speech recognition experiments are performed. The speech recognition system uses HMMs and Gaussian mixture models for acoustic modeling. No acoustic adaptation is used. The decoder to which the pruning is implemented is the one–pass time synchronous decoder[6]. Training corpus consists of 3696 English sentences from a large vocabulary named American English Corpus (TIMIT); test corpus is recorded by a speaker, 100 sentences extracted from TIMIT at random. Speech sampling frequency is 16KHz .Window function is the hamming of 25ms, and frame length is 10ms. Feature vector contains 39 dimensions, especially, and the basic feature vector consists of 12-dim Mel cepstral dimension with 1-dim short-term energy[7].

Test platform is a PC with Intel Pentium 4 with 2.99GHz processor and a 1GB RAM.

The experiment is carried out in a closed lab. System recognition accuracy calculated by the following formula

$$Acc\% = (W - S - D - I)/W \quad (7)$$

where W , T , D and I denotes total words, substitution errors, delete errors and insert errors respectively.

The traditional Viterbi-Beam Algorithm pruning threshold is $b(w)$=160 and $b(s)$=200.

TABLE I. RECOGNITION PERFORMANCE FOR DIFFERENT α

α	Recognition time (s)	recognition rate (%)
0.9	37	79.14
0.8	44	81.11
0.7	50	83.44
0.6	56	87.82
0.5	90	87.82

TABLE II. COMPARISON BETWEEN THE TWO ALGORITHMS

search algorithm	recognition time (s)	recognition accuracy (%)
Viterbi-Beam	90	87.82
Adaptive Viterbi-Beam	56	87.82

Table 1 shows that the value of α has a significant impact on system performance. During the process of α decreasing from 0.9 to 0.6, more paths are kept in phase one and the recognition accuracy increases. When α decreasing to 0.5, although some new paths will be kept, they hardly become the optimal path and the accuracy doesn't change. Obviously, the optimal value of α is 0.6.

Table 2 shows that the proposed pruning criteria can reduce the search space over 37.78% without compromising the recognition accuracy.

V. CONCLUSION

In this paper, we analyze the traditional Viterbi-Beam search algorithm and propose an improved adaptive Viterbi-Beam search algorithm. This method adaptively adjust the pruning threshold in initial phase and keep the value in phase two and phase three. The experimental result shows that the search space is compressed effectively without affecting the recognition accuracy. It has a certain reference value to real-time voice system. In addition, laws of active model number in the second phase and the third phase three need to be further researched to improve recognition accuracy.

REFERENCES

[1] X. L. Aubert, "An overview of decoding techniques for large vocabulary continuous speech recognition," Computer Speech and Language, vol.16, No.2, pp. 89-114, 2002

[2] J. J. Odell, "The use of context in large vocabulary speech recognition," PhD dissertation of University of Cambridge, UK, 1995

[3] Cui Yu-jing, "Strategies for improving efficiency of Speech Recognition," Master thesis of Beijing University of Posts and Telecommunications, 2008

[4] C. J. Hwang, S. J. Oh, and H. Y. Jung, "An adaptive pruning threshold algorithm for efficient Speech Recognition," PhD dissertation of Yeungnam University, 1999.

[5] J. Pylkkonen, "New Pruning Criteria for Efficient Decoding," 9th European Conference on Speech Communication and Technology, September, 2005, pp.581-584

[6] J. Pylkkonen, "An efficient one-pass decoder for Finnish large vocabulary continuous speech recognition," 2nd Baltic conference on human language technologies, 2005, pp.167-172

[7] L. R. Rabiner, R. W. Schafer, "Introduction to Digital Speech Processing," Now Publishers, Hanover, MA, USA, 2007