ONE-PASS COARSE-TO-FINE SEGMENTAL SPEECH DECODING ALGORITHM

Yun Tang[1*], *Wen-Ju Liu*[1], *Hua Zhang*[1], *Bo Xu*[1] *and Guo-Hong Ding*[2]

[1]National Laboratory of Pattern Recognition,Institute of Automation, Chinese Academy of Sciences
[2]Nokia Research Center, Beijing
{tangyun,lwj,hzhang}@nlpr.ia.ac.cn, xubo@hitic.ia.ac.cn, guohong.ding@nokia.com

ABSTRACT

In this paper, a novel one-pass coarse-to-fine decoding algorithm is proposed to accelerate the speed of Segment Model (SM). The algorithm is originated from the segmentation similarity observation described in the paper and is specific for the SM based speech recognition. At each step, a coarse search is first implemented to get coarse segmentations and then a fine search is performed based on the derived segmentation information. This fast algorithm is successfully integrated into an SM based Mandarin LVCSR system and saves more than 50% decoding time without obvious influence on the recognition accuracy.

1. INTRODUCTION

Segment model (SM) is a family of methods that adopt segmental distribution rather than frame-based features to represent the underlying trajectory of the observation sequence. So some limitations of Hidden Markov Model(HMM), such as the observations conditional independent assumption and non-stationary observation sequences that are modelled by a piecewise constant state [2][8], can be partially resolved by SMs. Due to the high complexity, SMs are hard to be employed in large vocabulary continuous speech recognition (LVCSR) unless the speech utterance is pre-segmented by HMM[9][10]. How to directly and efficiently apply SMs into the LVCSR system is still an open issue.

V. Digalakis et al [3] suggested a pruning method to speed up SMs by estimating segment scores from part of segments. In our previous work, a coloring speech method[5] is used to choose acoustic models before measuring segment scores. These methods decrease the run-time of SMs greatly and have laid the foundation for the application of SMs in LVCSR. However, the run-time of the SM based system is still slower than the real-time and more fast algorithms are required.

In this paper, a novel segmental based coarse-to-fine(CF) algorithm is proposed. It consists of two phases, a coarse extension phase and a fine extension phase. These two phases are applied at each step one by one. The coarse phase decodes the speech in a jumping way while the fine phase refines the decoding by means of measuring these points which are important for the final recognition result. These key points are detected in the coarse phase and the extension results obtained in the fine phase are the basis for following coarse searches. Both phases are carried out in one-pass while the previous coarse-to-fine decoding methods are multi-pass based [4].

The rest of this paper is organized as follows. A brief introduction of Stochastic Segment Model(SSM)[2], the acoustic model adopted in the system, is given in Section 2. Then in Section 3, the CF algorithm is presented. Section 4 shows the experimental results and analysis. Conclusions are drawn in the last section.

2. STOCHASTIC SEGMENT MODEL

SSM represents the variable length observation sequence by a fixed length region sequence. A resample function is needed to map the observation segment $x_1^N = \{x_1, x_2, ...x_N\}$ to the fixed length frame sequence y_1^L. The re-sampled frame is measured by "region", which is similar to the conception of the state in HMM, and L is the length of region sequence in each SSM.

$$y_i = x_{\lfloor \frac{i}{L} N \rfloor}, 0 < i \leq L, \tag{1}$$

where $\lfloor z \rfloor$ is the maximum integer no larger than z.

The log-likelihood of a segment x_1^N given model α:

$$\ln[p(x_1^N|\alpha)] = \sum_{i=1}^{L} \ln[p(y_i|\alpha, r_i)] \tag{2}$$

where r_i is the i-th region model in segment model α. Usually, each region consists of mixture Gaussians.

The decoding of SSM is a two-level process [1]. The first level is segment classification by the Bayesian approach.

$$\begin{aligned} D_m(\tau) \ = \ & \max_{\alpha}\{\ln[p(x_\tau^m|\alpha)](m-\tau) + \ln[P(\alpha)] \\ & + \ln[P_s(x_\tau^m|\alpha)]\}, \ \ 0 \leq \tau < m. \end{aligned} \tag{3}$$

where $D_m(\tau)$ is the highest likelihood score for feature segment x_τ^m, $P(\alpha)$ is the language score and $P_s(x_\tau^m|\alpha)$ is the

[*]This work was supported in part by the China National Nature Science Foundation (No. 60172055, No. 60121302) and the Beijing Nature Science Foundation (No.4042025).

segmental score (duration, etc). Then the combination of segmentations with highest probability is chosen as the recognition result in the second level. Given a sentence x_1^T, the process can be expressed as follows:

$$J^*(m) = \max_{\tau}\{J^*(\tau) + D_m(\tau) + C\},\ J^*(0) = 0, \qquad (4)$$

$$Bls(m) = \arg\max_{\tau}\{J^*(\tau) + D_m(\tau)\},\ Bls(0) = 0, \qquad (5)$$

where $1 \le m \le T, 0 \le \tau < m$, $Bls(m)$ is the best local start (BLS) point for m, $J^*(m)$ is the accumulated score of the best acoustic model sequence at point m and C is the insertion factor for each segment. A candidate set and an expanding set are formed at each point during decoding. The candidate set is a collection of hypothesized models ending at this point and the expanding set is the collection of triphone models which succeed the models in the candidate set. The decoding is performed from 1 to T point by point to get the final solution.

3. THE CF DECODING ALGORITHM

3.1. Segmentation similarity

Segmentations with high likelihood are always similar to or differ from the "true" segmentation by one or two frames. The "true" segmentation can be the result of recognition or force-alignment. A good demonstration for this is the n-best list or word lattice, in which paths with high likelihood have similar structure.

Due to the similarity of segmentation, we can "guess" the underlying boundaries from partial segmentation information. Based on the observation above, a segment based CF algorithm is developed, which includes two phases at each step: a coarse extension phase and a fine extension phase.

3.2. Coarse extension

Instead of estimating all possible partitions, only segments that begin and end at the basic point set $\mathcal{K}_S = \{0, S, 2S, ...\}$ will be measured in the coarse phase, where S is the CF step. $J_c^*(m)$ and $\hat{Bls}(m)$ are defined as the approximate versions of $J^*(m)$and $Bls(m)$ in $\mathcal{K}_S$,

$$J_c^*(m) = \max_{\tau}\{J_f^*(\tau) + D_m(\tau) + C\}, \qquad (6)$$

$$\hat{Bls}(m) = \arg\max_{\tau}\{J_f^*(\tau) + D_m(\tau)\},\ \ m, \tau \in \mathcal{K}_S, \qquad (7)$$

The definition of J_f^* is a little different from J^* in Equ.(4) and will be given in the fine extension phase of CF later. For examples, when S equals 1, the CF algorithm is simplified to the conventional SM decoding algorithm; when S sets to 2, only segments begin and end at even points will be measured in the coarse phase. No losing generality, we assume that T, the last frame of the speech sequence, belongs to $\mathcal{K}_S$. For those sequences incompatible with this assumption, we can add some silence frames at the end of the sequence and it will not influence the recognition accuracy.

3.3. Fine extension

The fine extension phase is to compensate the accuracy loss introduced during the coarse phase. We define the neighboring points of $\kappa \in \mathcal{K}_S$ as follows:

$$Nr(S, \kappa) = \begin{cases} \emptyset, & \kappa = 0 \\ \{\kappa{-}S{+}1, \ldots, \kappa{+}S{-}1\} - \{\kappa\}, & \kappa \ge S \end{cases} \qquad (8)$$

In the coarse phase, the decoder only stops at $\mathcal{K}_S$. Assuming the decoder currently moves to point $m \in \mathcal{K}_S$ and we get $\hat{Bls}(m)$ during the coarse extension phase. Based on the segmentation similarity introduced in Section3.1, we assume:

$$Bls(m') \in Nr(S, \hat{Bls}(m)) \cup \{\hat{Bls}(m)\}, \qquad (9)$$

where $m' \in Nr(S, m) \cup \{m\}$. So measuring segments originated from points in$Nr(S,\hat{Bls}(m))$ to m will be helpful to obtain a better BLS point of m. It is called the current-point extension (CPE). Segment models extended to m during the coarse phase and the fine phase will form the candidate set at m together. In CPE phase, J_f^* is defined as

$$J_f^*(m) = \max_{\tau}\{J_f^*(\tau) + D_m(\tau) + C\}, \tau \in \mathcal{K}_S \cup Nr(S, \hat{Bls}(m)). \qquad (10)$$

Such extensions may not be available for those points, of which $\hat{Bls}$ points are larger than S, since no hypothesized model will end at the neighboring points of those $\hat{Bls}$ points in the coarse extension. Hence, another extension, the fore-point extension (FPE), is introduced to connect those points, i.e. $Nr(S, \hat{Bls}(m)$ in Equ.(10), with the obtained decoding result. According to the assumption in (9), the BLS points for points in $Nr(S, \hat{Bls}(m)$ are limited to $Nr(S, \hat{Bls}(\hat{Bls}(m)))$. So, we can measure segments from $Nr(S, \hat{Bls}(\hat{Bls}(m)))$ to $Nr(S, \hat{Bls}(m)$ to get the J_f^*s of $Nr(S, \hat{Bls}(m))$. And, J_f^* in Equ.(10) is re-defined as Equ.(11) in FPE,

$$J_f^*(\tau) = \max_{\tau'}\{J_f^*(\tau') + D_\tau(\tau') + C\}, \tau \in Nr(S, \hat{Bls}(m))$$
$$\tau' \in Nr(S, \hat{Bls}(\hat{Bls}(m))) \cup \{\hat{Bls}(\hat{Bls}(m))\}, \qquad (11)$$

Again, the validation for point τ' in Euq.(11) needs to be considered.

1. If $\hat{Bls}(\hat{Bls}(m)) = 0, Nr(S, \hat{Bls}(\hat{Bls}(m))) = \emptyset$ and $J_f^*(0)$ set at the initial stage of the algorithm;

2. If $\hat{Bls}(\hat{Bls}(m)) \ge S$, $m > \hat{Bls}(m)$ and the CPE at $\hat{Bls}(m)$ has been done. So $J_f^*(\tau')$s were obtained when the CPE was done at $\hat{Bls}(m)$.

In the both cases, $J_f^*(\tau')$s have already been measured before we do FPE at m, so the FPE can be done successfully.

The whole CF procedure is illustrated by the flow chart in Fig.1. When the decoder moves to the current point m, a

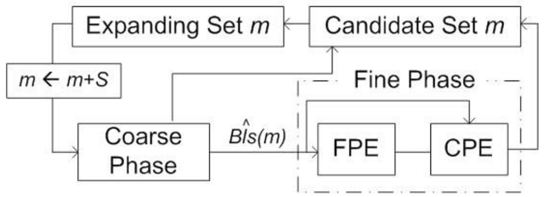

Fig. 1. *The flow chart of CF decoding.*

```
$m \leftarrow 0, J_0^* \leftarrow 0$;
while $m \leq T$ do
    $m \leftarrow m + S$
    $m_{min} \leftarrow \max\{m - L_{ext}, 0\}$
    $\tau \leftarrow \max\{(m - S), 0\}$
    while $\tau \geq m_{min}$ do                    // Coarse Phase //
        measuring $D_m(\tau)$, $\tau \leftarrow (\tau - S)$
    repeat
    computing $J_c^*(m)$ and $\hat{Bls}(m)$
    foreach $k \in Nr(S, \hat{Bls}(m))$               // Fine Phase //
        if $J_f^*(k)$ is not available, then
            foreach $d \in Nr(S, \hat{Bls}(\hat{Bls}(m))) \cup \{\hat{Bls}(\hat{Bls}(m))\}$
                measuring $D_k(d)$                     // FPE //
            repeat
            computing $J_f^*(k)$ by Equ.(11)
        endif
        measuring $D_m(k)$                             // CPE //
    repeat
    computing $J_f^*(m)$ by Equ.(10)
repeat
Trace back to get the best word sequence. #
```

Fig. 2. *The CF Extension algorithm of SM. L_{ext} is the allowed maximum segment duration.*

coarse extension is executed; then we do CPE at the neighboring points of $\hat{Bls}(m)$ to compensate the accurate loss introduced by the jumping decoding. If $J_f^*(\tau)$s have not been obtained, where $\tau \in Nr(\hat{Bls})$, a FPE is required for these points before the CPE is done at m. Finally, the decoder moves forward from m to $m + S$ and the procedure will be repeated until the final point is reached. Details of the algorithm are listed in Fig.2.

4. EXPERIMENTS AND ANALYSIS

4.1. Experimental setup

The data corpus applied in experiments is provided by Chinese National Hi-Tech Project 863 for Mandarin LVCSR system development [6]. 83 male speakers' data are employed for training (48373 sentences, 55.6 hours)and 6 male speakers' for test (240 sentences, 17.1 minutes). Acoustic features are 12 dimensions MFCC plus 1 dimension normalized energy and their 1st and 2nd order derivatives. There are 24 syllable initials and 37 syllable finals in our Mandarin phoneset. Each syllable final has 5 tones.

Table 1. *Parameters of HMM and SSM in system.*

Model	Models	Regions	Mixtures	Durations
HMM	18364	5068	16	–
SSM	24180	7983	12	2144

Table 2. *Comparison of HMM and SSM for Test-863.*

Model	Sub%	Del%	Ins%	Err%
HMM	15.6	0.1	1.3	17.0
SSM	12.9	0.1	0.0	13.0

Table 3. *Recognition results with different CF step S. "$*$" rows are the results without the fine phase.*

S	Sub%	Ins%	Del%	Err%
2	12.8	0.2	0.0	13.0
3	13.7	0.2	0.0	13.9
2^*	15.0	0.2	0.0	15.2
3^*	16.9	0.2	0.1	17.1

The baseline of SM is a context-dependent triphone SSM recognition system [5]. The search pathes are organized by the lexical tree and begin/end with the silence model. Each segment model is sequentially composed of 15 regions and each region is modelled by 12 Gaussian mixtures. Region models are tied by phone based decision trees. Triphone based duration models are used to improve the system accuracy.

To make the experiments comparable, a continuous density HMM (CDHMM) is developed as the baseline of HMM by HTK V3.2.1 [7]. The structure of HMM is left to right with 5 states, 3 emitting distributions and no state skipping, except "sp" (short pause) model with 3 states, 1 emitting distribution. Each emitting distribution is modelled by 16 Gaussian mixtures. The details of these two baselines are listed in Table 1. The "Models", "Regions", "Mixtures" and "Durations" in Table 1 are the number of models, regions (or states), mixtures and duration models in SSM (or HMM) respectively. A bigram language model with 48188 words is used both in HMM and SSM systems.

4.2. Results and analysis

Table 2 gives the baseline recognition results of the test set. In this table, "Del", "Ins", "Sub" and "Err" represent the deletion, insertion, substitution and character error rate respectively. The HMM baseline achieves 17.0% character error rate and SSM achieves 23.5% relative error reduction compared with the HMM baseline. The SSM result is even comparable to [6], in which a trigram language model is used.

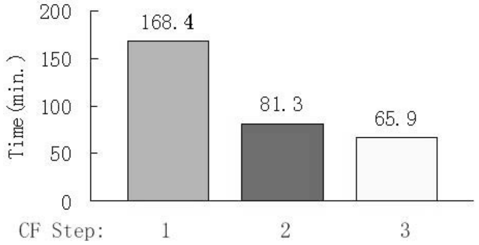

Fig. 3. *The run-time as function of the CF step S.*

Table 3 gives the recognition results with different CF steps. When S sets to 2, the recognition result is slightly changed compared with the original one; when S equals 3, the recognition result is downgrade slightly. Rows with ”*” show the recognition results of the CF algorithm without the fine phase. The error rate is increased obviously when the fine search step is ignored. The time comparison is illustrated in Fig.3. The decoding time without CF is 168.4 minutes and is approximately 10 times of the real time. When S sets to 2, 51.7% decoding time is saved; and more than 18.9% time is saved again when S equals 3. Since the run time of HMM with language model built by HTK is much slower than real systems, we don’t compare it with the SSM system.

4.3. Complexity analysis

We measure the complexity of the decoding algorithm from two aspects: the number of segments extended and the number of Gaussian models measured. For a sentence with T frames, the maximum segment duration is L_{ext} and the average number of hypothesized models expanded at each points is M. That is to say, there are average $L_{ext}M$ model instances expanded to the current point and the number of model instances evaluated per sentence using the conventional decoding is approximately $O(T \cdot L_{ext} \cdot M)$. The CF algorithm only needs to compute $L_{ext} \cdot M/S + 4S \cdot (S-1) \cdot M$ instances at each time point, where $L_{ext} \cdot M/S$ instances are expanded in the coarse extension phase, $2(2S-1) \cdot (S-1) \cdot M$ instances in FPE and $2(S-1) \cdot M$ instances in CPE. Considering that most BLSs for points in the same segment are same, FPE is not necessary in most points and the number of model measured during fine phases is much less than the one in coarse phases. If we ignore the number of instances computed in fine phases, we could get the bottom boundary of model instances evaluated per sentence, that is, $O(T \cdot L_{ext} \cdot M/S^2)$, which is S^2 times less than the one without CF. Due to region models could be shared by different model instances (different models or same model with different durations), it is hard to give a general formula for the number of Gaussian models evaluated during decoding. In practice, when S sets to 2, the number of Gaussian models evaluated in the improved method is average 15.2% less than the conventional one’s (216150 vs. 254894 per sentence).

5. CONCLUSIONS

In this paper, A novel SM decoding method, the one-pass CF algorithm, is proposed to decrease the aimless extensions during SM decoding. The coarse phase offers useful hints for the successive fine phase to expand models and the fine phase gives a precise basis for the following coarse phases. The two phases are performed alternately at each basic point and the search process is finished within one pass. This fast algorithm greatly enhances an SSM-based LVCSR system without obvious influence on the performance, which is significantly better than the corresponding HMM-based system. The experiments show that, after fast SM algorithms have been employed, SSM could be a good alternative to HMM for LVCSR.

6. REFERENCES

[1] M. Ostendorf, V. Digalakis and O. Kimball, ”From HMM’s to Segment Models: A Unified View of Stochastic Modeling for Speech Recognition,” *IEEE Trans. Speech and Audio Proc.*, 1996, pp. 360-378.

[2] M. Ostendorf, S. Roukos, ”A stochastic segment model for phoneme based continuous speech recognition,” *IEEE Trans. Speech and Audio Proc.*, 1989, pp.1857-1869.

[3] V. Digalakis, M. Ostendorf, J. Rohlicek, ”Fast Algorithms for phone classification and recognition using Segment-based Models,” *IEEE Trans. Speech Audio Proc.*, 1992, pp. 2885-2896.

[4] X. Huang, A. Acero, and H. Hon, ”*Spoken Language Processing: A Guide to Theory, Algorithm and System Development*,” Prentice Hall PTR, April 25, 2001.

[5] Y. Tang et al, ”Coloring the Speech Utterance to Accelerate the SM based LVCSR Decoding”, in *Proc. NLP-KE*, Wuhan, China,Oct. 30- Nov.1, 2005.

[6] S. Gao, et al, ”Update of Progress of Sinohear: Advanced Mandarin LVCSR System At NLPR”, in *Proc. ICSLP*, Beijing, 2000.

[7] S. Young et al, *The HTK Book*, Cambridge, 2000.

[8] H.W. Hon and L. Wang, ”Combining frame and Segment based models for LVCSR”, *IEEE workshop on ASRU*, Keystone, 1999.

[9] J. Glass, ”A probabilistic framework for segment-based speech recognition,” *Computer Speech and Language*, 17 (2003), pp. 137-152.

[10] S. Au Yeung, C.F. Li and M.H. Siu, ”Sub-phonetic polynomial segment model for large vocabulary continuous speech recognition,” in *Proc. ICASSP*, Philadelphia, 2005.