TIME-SYNCHRONOUS CONTINUOUS SPEECH RECOGNIZER DRIVEN BY A CONTEXT-FREE GRAMMAR

Tohru Shimizu, Seikou Monzen†*, Harald Singer and Shoichi Matsunaga*

ATR Interpreting Telecommunications Research Labs.,
2–2, Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-02, Japan
shimizu@itl.atr.co.jp
†Yamagata University

ABSTRACT

This paper proposes a time-synchronous continuous speech recognizer driven by a context-free grammar that integrates generalized LR parser based phoneme context prediction and context-dependent HMMs. In this method, a phoneme hypotheses trie is introduced for the phoneme history representation of possible LR states, and an LR state network is introduced for LR path merging. Both techniques reduce the amount of computation. The experimental results show that this new method is more efficient than the conventional LR parser driven phoneme-synchronous continuous speech recognizer.

1. INTRODUCTION

A generalized LR (GLR) parsing algorithm[1] provides an effective framework for combining linguistic information and phonological information. Phoneme-synchronous search which uses GLR parsing and HMM phoneme verifiers has been proposed as a continuous speech recognizer for phrase-wise spoken utterances(HMM-LR[2], SSS-LR[3]). Figure 1 shows the schematic diagram of the conventional phoneme-synchronous HMM-LR (PS-HMM-LR) method. However, this method has the following two serious problems:

1. **Redundant calculation of acoustic scores for hypotheses with the same phoneme history but a different syntactic parse**
   This problem is due to the use of a syntactic state oriented hypothesis *cell*. As each *cell* contains the information for one possible parse, many *cells* are created for the same phoneme sequence.
   (e.g. Two different *cell* hypotheses "hada(noun)" and "ha(noun),da(copula)" are created for a single phoneme sequence "hada". In other words, the acoustic matching is done twice for the same phoneme history "hada".)
2. **Decrease in acoustic accuracy caused by the normalization of probabilities for beam search**
   This problem is inherent in phoneme-synchronous search because hypotheses have different time lengths. This requires the time normalization of acoustic probabilities to compare the scores of different length.

In addition, the computational cost of the PS-SSS-LR method increases exponentially as the input speech becomes longer. On the other hand, the computational cost for one-pass search is proportional only to the number of hypotheses [4][5][6].

This paper proposes a time-synchronous continuous speech recognizer driven by an LR parser. Three data structures, i.e., (1) *grid* hypotheses independent of syntactic parse, (2) a trie [7] structured phoneme history (phoneme hypotheses trie), and (3) an LR state network, are introduced to solve the first problem, while time-synchronous search is introduced to solve the second problem.

2. SYSTEM STRUCTURE

Figure 2 shows the schematic diagram of the time-synchronous HMM-LR (TS-HMM-LR) method consisting of one-pass Viterbi search and LR parser driven phoneme context prediction.

The data structure "*grid*" is introduced as a hypothesis representing the phoneme history and its acoustic score. The *grid* hypotheses are defined by (i, j, n, s), where

i	input frames
j	HMM state
n	predicted phoneme context (phoneme triplet)
s	mode of phoneme hypotheses trie

The merit of a *grid* hypothesis is that it is independent of syntactic parse and is time-synchronously updated by one-pass Viterbi search.[1] In addition, acoustic score normalization is not necessary for a *grid* hypothesis because the time-lengths of all hypotheses are the same. Accordingly, the second problem is solved.

To solve the first problem, an LR parser driven architecture requires an efficient interface between the *grid* hypotheses and the syntactic parse. In addition, syntactically ambiguous parses for the same phoneme history have to be merged. In the TS-HMM-LR

[1] By contrast, a *cell* hypothesis in the PS-HMM-LR method is dependent on the syntactic parse and is phoneme-synchronously updated.

0-7803-2431-5/95 $4.00 © 1995 IEEE

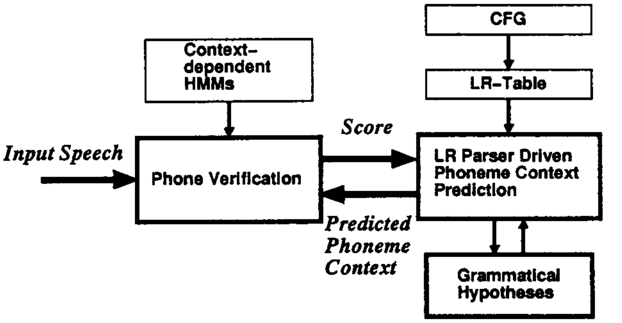

Figure 1. A conventional phoneme-synchronous CSR driven by a CFG (PS-HMM-LR method)

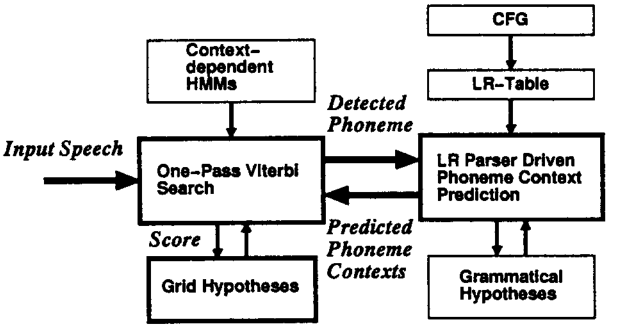

Figure 2. A proposed time-synchronous CSR driven by a CFG (TS-HMM-LR method)

method, this interface function is achieved by predicted phoneme context merging of the possible syntactic parses with the same phoneme history.

The following section describes the predicted phoneme context merging algorithm introduced in the LR parser driven phoneme context prediction module.

3. PREDICTED PHONEME CONTEXT MERGING OF POSSIBLE SYNTACTIC PARSES

Two new data structures (i.e., a phoneme hypotheses trie and an LR state network) are introduced to enable the predicted phóneme context merging.

Phoneme hypotheses trie
A trie-structured phoneme history representation. A phoneme history $H(s)$ and corresponding LR state network nodes $P(s)$ are identified by the trie node s.

LR state network
A graph-structured LR stack representation. Each node and arc of LR stack network express the LR state and action, respectively. The LR path merging is realized as a transition to the same network node.

A phoneme context prediction algorithm using a phoneme hypotheses trie and an LR state network proceeds as follows:

```
if there already exists phoneme trie node $s'$
then do nothing
else create phoneme trie node $s'$ using $s$ and $n$
    for each network node $p \in P(s)$
        for each action $a$ on node $p$
            if there already exists an arc to node $q$ for
                    action $a$ from node $p$
            then do nothing
            else create new network node $q$
                    get predicted phoneme context $C_g(q)$
                        by looking up LR-table
            endif
            $P(s') \leftarrow P(s') \bigcup q$
            $C_t(s') \leftarrow C_t(s') \bigcup C_g(q)$
        end
    end
endif
return $C_t(s')$
```

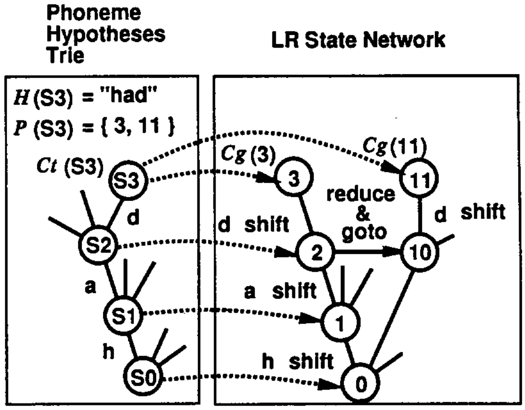

Figure 3. A simple example of the phoneme context merging using a phoneme hypotheses trie and an LR state network

Figure 3 shows a simple example of the phoneme context merging using an LR state network and a phoneme hypotheses trie. When a phoneme /d/ is detected in the search module, a new phoneme hypotheses trie node S_3 is created ($H(S_3)$ = *"had"*) and corresponding LR state network nodes are also created ($P(S_3) = \{3, 11\}$). The predicted phoneme context for each LR state network node ($C_g(3), C_g(11)$) is calculated by looking up the LR table, and the union of predicted phoneme contexts is stored in the phoneme hypotheses trie node S_3 ($C_t(S_3) \leftarrow C_g(3) \bigcup C_g(11)$)[2].

Thus, all syntactic parses with the same phoneme history are merged in the same phoneme hypotheses trie node. As the search module only accesses the phoneme hypotheses trie, syntactic ambiguity does not

[2] Both predicted phoneme context $C_g(q)$ and $C_t(s')$ are phoneme triplets because a context-dependent HMM is used.

influence expansion of the *grid* hypotheses. Accordingly, the first problem does not occur in the TS-HMM-LR method.

4. RECOGNITION EXPERIMENTS

4.1. Experimental conditions

Experiments evaluating phrase recognition and sentence recognition were carried out using the dialogue database on a conference registration task[8]. Context-dependent HMMs (HMnet[9]) were used in these experiments. For the speaker-independent experiment, an HMnet was composed using a composition and clustering method of speaker-dependent HMMs[10]. Other experimental conditions are summarized in Table 1.

Table 1. Experimental Conditions

Analysis conditions	
Sampling rate	12 kHz
Window	Hamming window (20 ms)
Frame period	5 ms
Analysis	log power + 16-order LPC-Cep + Δlog power + 16-order ΔLPC-Cep
CFG	
Rules	978 (intra-phrase grammar) 1009 (sentential grammar)
Words	457
Phoneme perplexity	2.66 (phrase test data) 2.79 (sentence test data)
HMnet(speaker-dependent)	
State	600 states
Training	2620 Japanese words
HMnet(speaker-independent)	
State	600 states (15 mixture)
Training	285 speakers (145 male, 140 female) 50 Japanese sentences
Test data	
Phrase data	1 male 701 Japanese phrases
Sentence data	2 male, 2 female 136 sentences

4.2. Speaker-dependent phrase recognition test by using PS-HMM-LR method and TS-HMM-LR method

Speaker-dependent phrase recognition tests were performed to compare the efficiency of the TS-HMM-LR and the conventional PS-HMM-LR methods. All of the experiments were run on an HP9000/735 workstation. The beam-search technique was used in these experiments. As the beam width of the TS-HMM-LR method is not comparable with that of the PS-HMM-LR method, the CPU time and recognition rate were used to compare efficiency. Figure 4 shows the CPU time requirement of the TS-HMM-LR method for various beam widths normalized by that of the PS-HMM-LR method for a fixed beam width. In this experiment, the TS-SSS-LR method with a beam width of 235 was expected to consume the same amount of CPU time as the PS-SSS-LR method with a fixed beam width of 250.

Figure 5 shows phrase recognition error rates for both methods. For the same CPU time requirement, the error rates of the TS-HMM-LR method and PS-HMM-LR method are 6.1% and 9.3%, respectively. These results show that the TS-HMM-LR method achieved a higher accuracy than the PS-HMM-LR method by avoiding redundant acoustic calculations and by using acoustic probabilities without normalization.

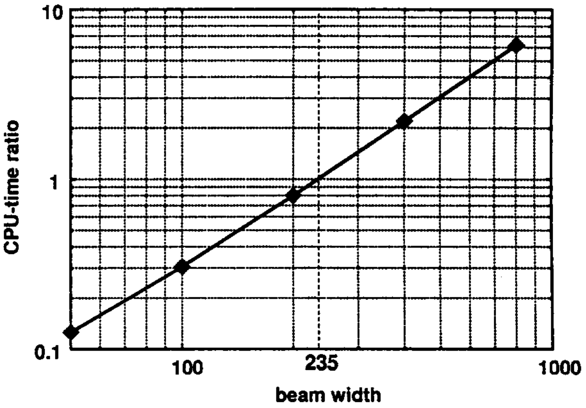

Figure 4. Ratio of CPU time requirement of the TS-HMM-LR method for various beam widths (50-800) normalized by that of the PS-HMM-LR method for a fixed beam width (250)

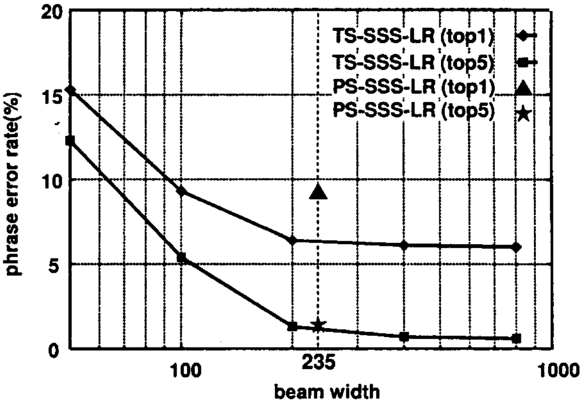

Figure 5. Phrase error rate reduction by using the TS-HMM-LR method compared with the phrase error rate of the PS-HMM-LR method for the same CPU-time requirement (speaker:male1)

4.3. Speaker-dependent sentence recognition test by using TS-SSS-LR method

Sentence recognition tests were carried out using the TS-HMM-LR method. The sentence error rates are shown in Figure 6. In this experiment, the TS-HMM-LR method achieved a sentence error rate of 29.4% for the top candidate and 17.6% for the five best candidates with a beam width of 3200.

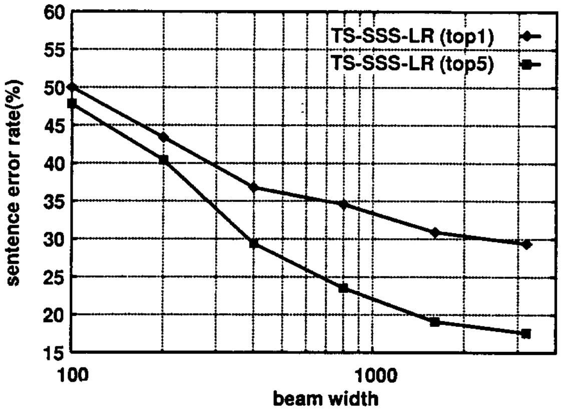

Figure 6. Sentence recognition error rate for the TS-HMM-LR method with various beam widths (100-3200) (speaker:male1)

4.4. Speaker-independent sentence recognition test by using TS-SSS-LR method

Table 2 shows the sentence error rates for the top candidates using speaker independent HMMs for various speakers. The beam width was 3200. The error rate for the same speaker (male1) increases 15.0 % compared with that obtained using speaker-dependent HMMs. The average sentence error rate of the top candidates is 42.3 % for male and 49.1 % for female speakers.

Table 2. Sentence error rates using speaker-independent HMMs (beam width 3200)

Speaker	sentence error rate (%)		
	top1	top5	top10
male1	33.8	22.8	22.1
male2	50.9	44.7	40.2
female1	40.1	25.9	25.9
female2	58.1	37.5	37.5

5. CONCLUSION

This paper has proposed a new time-synchronous continuous speech recognition method (TS-HMM-LR) that integrates one-pass search and LR parser based phoneme prediction and context-dependent HMMs. In this method, a phoneme hypotheses trie was introduced for the phoneme history representation of possible LR states and an LR state network was introduced for LR path merging. The experimental results show that this new method requires much less computation than the phoneme-synchronous HMM-LR (PS-HMM-LR) method. However, a considerably larger beam width is still required for sentence recognition compared to phrase recognition. Furthermore, sentence error rates with speaker-independent HMMs are approximately 46 % larger than with speaker-dependent HMMs. To cope with these problems, we plan to incorporate statistical language models, such as n-gram and stochastic CFG.

ACKNOWLEDGMENTS

The authors are deeply grateful to Dr. Yamazaki, president of ATR Interpreting Telecommunications Research Laboratories, and to all members of the laboratory for their help and encouragement.

REFERENCES

[1] M. Tomita: "An Efficient Context-free Parsing Algorithm for Natural Languages", Proc. of IJCAI85, pp.756-763, (1985).

[2] K. Kita, T. Kawabata and H. Saito: "HMM Continuous Speech Recognition using Predictive LR Parsing", Proc. of ICASSP89, pp.703-705, (1989).

[3] A. Nagai, J. Takami and S. Sagayama: "The SSS-LR Continuous Speech Recognition System: Integrating SSS-Derived Allophone Models and a Phoneme-Context-Dependent LR parser", Proc. of ICSLP92, pp.1511-1514, (1992).

[4] J. S. Bridle, M. D. Brown, and R.M. Chamberlain: "An Algorithm for Connected Word Recognition", Proc. of ICASSP82, pp.899-902, (1982).

[5] H. Ney, D. Mergel, A. Noll and A. Paeseler: "A Data-Driven Organization of the Dynamic Programming Beam Search for Continuous Speech Recognition", Proc. of ICASSP87, pp.833-836, (1987).

[6] H. Ney: "Dynamic Programming Parsing for Context-Free Grammars in Continuous Speech Recognition", IEEE Trans. on Signal Processing, 39, 2, pp.336-340, (1991).

[7] A. V. Aho, J. E. Hopcroft and J. D. Ullman: "Data Structures and Algorithms", Addison-Wesley, MA, pp.163-169, (1983).

[8] T. Ehara, K. Ogura and T. Morimoto: "ATR Dialogue Database", Proc. of ICSLP90, pp.1093-1096, (1990).

[9] J. Takami and S. Sagayama: "A Successive State Splitting Algorithm for Efficient Allophone Modeling", Proc. of ICASSP92, pp.573-576, (1992).

[10] T. Kosaka and S. Matsunaga: "Speaker-Independent Phone Modeling based on Speaker-Dependent HMMs' Composition and Clustering", Proc. of ICASSP95, (1994).