IMPROVEMENTS IN BEAM SEARCH FOR 10000-WORD CONTINUOUS SPEECH RECOGNITION

H. Ney, R. Haeb-Umbach, B.-H. Tran, M. Oerder

Philips Research Laboratory Aachen
P.O. Box 1980, D-5100 Aachen, Germany

ABSTRACT

This paper describes the improvements in a time synchronous beam search strategy for a 10000-word continuous speech recognition task. The improvements are based on two measures: a tree-organization of the pronunciation lexicon and a novel look-ahead technique at the phoneme level, both of which interact directly with the detailed search at the state levels of the phoneme models. Experimental tests were performed for 4 speakers on a 12306-word task. As a result of the above measures, the overall search effort was reduced by a factor of 17 without a loss in recognition accuracy.

1. INTRODUCTION

In this paper, we describe the improvements in a time synchronous beam search strategy for a 10000-word continuous speech recognition system. Like many other systems, this system is based on hidden Markov models, and therefore the search for the most probable word sequence is the computationally most expensive operation of the recognition system. In time-aligning the acoustic vectors of the input utterance with the reference models, a huge number of possible paths has to be considered as a result of the vocabulary size and the segmentation ambiguities in continuous speech. In [1], a dynamic construction of the search space along with a list organization of the active hypotheses had been introduced. This baseline system had been originally designed for and tested on several 1000-word tasks such as SPICOS database queries (German) [2] and DARPA resource management (US English) [3]. When extending this system from 1000 to 10000 words, we were faced with a rather high computational cost that was caused by the size of the search space and the time synchronous beam search strategy. After analyzing where the lion's share of the search effort was spent, we isolated three effects that allowed us to reduce the search effort by a factor of 17 without increasing the word error rate. These three effects are:

1. A tree organization of the lexicon reduces the search effort by a factor of 7 over the linear lexicon organization. This reduction factor is higher than the static compression factor of 2.5 between tree and linear lexicon, since most of the search effort is spent on the initial phonemes of the words.
2. Unlike a linear lexicon, a tree lexicon in conjunction with a bigram language model requires a tree copy for each of the 10000 predecessor words. Fortunately, however, the experimental results show that due to the beam search strategy, only a few tree copies are actually generated.
3. A look-ahead at the phoneme level within a tree was introduced to check every 10-ms time frame for each of the 44 generic phonemes whether the corresponding state hypotheses were likely to survive for the next 60 ms.

There have been a number of studies that used a tree organization or a fast match in various stages of the recognition process [4, 5, 6, 7]. The approach described in this paper has the following novel features: both the tree organization of the lexicon and the look-ahead technique are integrated into the beam search at the 10-ms level and are applied to continuous speech input.

After describing the baseline recognition system in Section 2, we present the tree structure for the lexicon and the search strategy in Section 3. Section 4 describes the phoneme look-ahead technique and its integration into the beam search pruning. In Section 5, systematic recognition experiments are reported on for 4 speakers.

2. RECOGNITION SYSTEM

The baseline recognition system is phoneme based and speaker dependent with a 12306-word (German) recognition vocabulary. There is a set of 44 'generic' context independent phonemes which are modeled by 6-state hidden Markov models. The emission distributions are modelled by mixtures of Laplacian densities. The system uses the Viterbi criterion, i.e. the most likely state sequence, both in training and in recognition. A full description of the details of acoustic-phonetic modelling can be found in [8].

The experimental results reported in this paper were obtained for the following database and conditions. The training is based on a set of 300 sentences comprising 2735 words. The recognition tests are based on a set of 50 sentences (German business correspondence) comprising 1099 words. There is only a small overlap between training and test vocabulary. All sentences were recorded for each of 4

0-7803-0532-9/92 $3.00

speakers using a hand-held microphone and analog tapes. The sentences were read, but there was no attempt to enforce careful pronunciation and high signal-to-noise ratio. The stochastic language model was either a bigram model or a unigram model of test set perplexities of 1056 and 1831, respectively.

When the 12306 words of the lexicon are expanded into phonemes and the phonemes into the states of the hidden Markov models, a search space of 650000 states is obtained which has to be processed every 10 ms, the spacing of the input time frames. The incorporation of a bigram language model into the recognition process does not affect the size of the search space [2]: the conditional probability of the word bigram under consideration is taken into account at the time the word boundary between the two words is hypothesized, and thus the language model recombination takes place at the beginning of a word hypothesis. To handle the huge search space and keep the search effort as small as possible, the following techniques are used [1]:

- *Beam search:* find the locally, i.e. at the current time frame, best state hypothesis and discard all state hypotheses that are less probable by more than a fixed threshold than the locally best hypothesis.
- *List organization of the active search space* [1]: introduce lists at several levels to dynamically construct the search space such that the computational cost of the search is now proportional only to the number of active hypotheses and is independent of the overall size of the potential search space.

Table 1 shows the distribution of the hypotheses over the phoneme positions for recognitions experiments on 4 speakers (M-21,-22,-23,-24); more details of the experiments are given in Section 5. The numbers are averaged over all 10-ms input frames for each speaker. The total number of state hypotheses per 10-ms frame is about 50000, which is only a small fraction of the total search space of 650000 states. The main search activity takes place at the beginning of the words: the first and second phoneme account for 79% and 16%, respectively, of the state hypotheses. Obviously, this effect is caused by the ambiguities in the word boundaries. Due to the left-to-right asymmetry in the search direction, there are on the average only about 10 state hypotheses per 10 ms that have reached the end of a word model.

Table 1: Distribution of the state hypotheses (x 1000) over the phonemes within a word.

Speaker	M-21	M-22	M-24	M-25	AVG.
All phonemes	40	42	50	50	100%
1st phoneme	34	34	34	39	79%
2nd phoneme	5	6	11	8	16%

3. TREE SEARCH

A large number of words in a 10000-word vocabulary begin with the same initial sequence of phonemes. Therefore it is advantageous to arrange the pronunciation lexicon as a tree. Each arc of the tree stands for a phoneme such that an arc sequence from the tree root to a tree leaf represents a legal phoneme sequence and thus a word of the vocabulary. The leaves mark the end of a word, and some of them may be located in the tree interior, since some words form the beginning part of another word. For the 12306-word vocabulary, Table 2 shows the distribution of arcs over the first 6 generation layers of the tree. The lexical tree consists of 43000 arcs, which is a compression factor of 2.5 over the 108000 phoneme copies of the linear lexicon. Table 2 also includes the average number of active arcs in the layers of the tree for a typical recognition experiment. Most of the search effort is still located in the first arcs of the tree. Thus we can expect that the increase in the search cost is less than proportional to the vocabulary size. A straightforward inclusion of a word look-ahead for further reducing the search cost is, however, not possible since the word identities are not known at the tree root.

Table 2: Distribution of tree arcs and active tree arcs over the layers of the tree lexicon.

Layer	1	2	3	4	5	6	≥ 7
Arcs	28	331	1511	3116	4380	4950	29200
Active	23	233	485	470	329	178	206

Unlike a linear lexicon, a tree lexicon requires a modification of the search space when a bigram language model is used. The reason is that, when a tree is started, all words are hypothesized and the word identity is only known at the end of the tree. Therefore the language model probabilities can only be incorporated after the end of the second word of the bigram has been reached. Thus the inclusion of language model scores may be viewed as delayed by one word as compared to the bigram model without using a tree lexicon. In particular, for a bigram language model, a tree copy is required for each predecessor word. Thus the potential search space is increased by a factor which is exactly the size of the vocabulary. The experimental results given in Section 5, however, show that due to the beam search strategy, only a few tree copies are required. As in the case of a linear lexicon [1], a careful organization of the search is required to avoid any computational overhead and guarantee a time complexity linear in the number of state hypotheses. The search process is organized in four parts: acoustic recombination, language model recombination, phoneme look-ahead, and search bookkeeping and traceback. The phoneme look-ahead will be explained in the next section.

For each input frame, the *acoustic recombination* module advances the hypotheses by one frame and performs the tree-interior time alignment, i.e. determines the best

within-word or within-tree state sequence. The implementation is based on list organizations at several levels: at the level of the vocabulary trees, the tree arcs and the phoneme states. The module reports a list of ending words to the *language model recombination* module which performs the recombination at the word level and returns a list of surviving word hypotheses to the acoustic recombination to start-up the corresponding trees. Note that the language model recombination is completely separated from the acoustic recombination. The only interface is the list of word hypotheses. The *bookkeeping and traceback* module keeps track of the word hypotheses such that at the end of the recognition, the recognized word sequence is recovered by chaining down the traceback arrays [1].

Table 3 shows the dependence of the search cost on the pruning threshold for a tree search based recognition with a bigram language model of test set perplexity P = 1056. A pruning threshold of 130 corresponds to five times the average log-likelihood per frame. In comparison with Table 1, the number of state hypotheses has been reduced from 40000 to only 3000-12000. Wheres the potential number of tree copies is 12306, only a very small fraction of these trees is actually active. Note that the number of state hypotheses varies extremely across a sentence. Although the average number of state hypotheses is only 7000 for a pruning threshold of 130, the average of the maximum number of state hypotheses per sentence (Avg_states) is ten times larger. The overall maximum number of state hypotheses per frame (Max_states) for the whole recognition experiment is even as high as 205000.

Table 3: Effect of pruning threshold on the number of search hypotheses for states (x 1000), arcs (x 1000) (B/A: before/after pruning) and trees and on the word error rate (speaker M-21; bigram language model with P=1056).

Threshold	110	120	130	140	150
Max_states	71	120	205	-	-
Avg_states	25	40	70	100	130
States:B	3	5	7	10	12
A	2	3	4	6	7
Arcs:B	1.0	1.5	2.5	3.5	4.5
A	0.7	1.0	1.7	2.5	3.0
Trees	6	11	18	25	30
Errors [%]	25.2	15.3	15.1	14.6	14.6

4. PHONEME LOOK-AHEAD

As discussed in the previous section, the number of state hypotheses can be considerably reduced by using the tree-organized lexicon and search. Nevertheless, the number of state hypotheses is still much larger than the number of generic states which is only 44 phonemes times 6 states per phoneme, i.e. 264 states. This is the number of state hypotheses that would be required in a full search for phoneme recognition. Due to the lexical and language model constraints, a huge number of phoneme copies has to be made. As a result, for each lexical tree, the number of state copies is 278000, and the average number of state hypotheses varies typically between 3000 and 10000.

The basic idea of the phoneme look-ahead is to estimate the probability of each phoneme ahead of the current time frame and to utilize this probability estimate in an additional pruning process of the tree search. This operation is only performed when a state sequence hypothesis across a phoneme boundary is encountered. Let t denote the index of the time frame under consideration. Let $S_D(t;q)$ denote the score (in negative logarithms) of the detailed match for the final segment of a phoneme copy q, i.e. the score of states from which the phoneme can be left. Let us consider one of the successor phonemes of q in the lexical tree and denote the look-ahead score for its generic phoneme Q by $S_L(t;Q)$. The additional pruning operation works as follows. A successor phoneme is activated in the search if and only if

$$S_D(t,q) + S_L(t,Q) < S_0 + min[S_D(t,q') : q'] + min[S_L(t,Q') : Q']$$

where S_0 denotes the pruning threshold of the tree search. In other words, phoneme hypotheses for which the state hypotheses are known to fall short of the pruning threshold at a later time frame will not be activated at the current time frame. This phoneme look-ahead is similar to the admissible word selection method described in [9]. Evidently, if the look-ahead estimate for the anticipatory time interval is correct, i.e. identical with the score of a detailed match, no additional search errors are introduced.

For the look-ahead, the detailed phoneme models are used. The log-likelihoods of the emission probabilities are stored and used later in the detailed match. The phoneme look-ahead produces the probability estimate for the best path extension beyond the current time frame by a time alignment. The best results were obtained for a window length of 60 ms for the anticipatory time interval. To reduce the amount of computation, a beam pruning strategy is also used in the time alignment of the phoneme look-ahead, which processes all generic phonemes in a time synchronous manner. Note that there is no need for backpointers and traceback arrays since the only objective is to obtain the look-ahead score for each phoneme. To further reduce the computational cost, the look-ahead is carried out only every other time frame. For the omitted time frame, the look-ahead scores of the previous time frame are used.

5. EXPERIMENTAL RESULTS

The experimental results are summarized in Table 4 for each of four speakers using four different recognition experiments (P = perplexity):

A: linear lexicon, bigram, P = 1056;

B: tree lexicon, unigram, P = 1831;

C: tree lexicon, bigram, P = 1056;

D: tree lexicon, bigram, P = 1056, phoneme look-ahead.

For each recognition experiment, Table 4 gives the search effort in terms of the average number of state hypotheses per 10-ms frame and the word error rate, namely deletion and insertion rate and total word error rate.

Table 4: Search effort [States/10-ms] and word error rates [%] for four recognition experiments (A,B,C,D).

	Speaker	Search effort	Word error rate [%]		
		[Sts/10-ms]	Del./Ins.	Total	Avg.
A	M-21	40000	1.5/2.4	14.6	
	M-22	42000	2.0/3.7	19.5	
	M-24	50000	4.2/3.1	27.1	
	M-25	50000	0.4/3.0	14.5	19.0
B	M-21	3200	2.8/2.5	19.8	
	M-22	4800	3.3/3.4	23.9	
	M-24	6400	6.0/2.7	31.6	
	M-25	6700	2.2/2.6	19.5	23.7
C	M-21	7000	1.7/2.3	15.1	
	M-22	9900	2.1/3.8	19.7	
	M-24	10300	4.5/2.5	25.8	
	M-25	8500	0.6/2.2	14.0	18.7
D	M-21	1850	2.1/2.3	15.4	
	M-22	2700	2.1/3.8	19.9	
	M-24	3500	4.6/2.5	26.0	
	M-25	3100	0.6/2.4	14.2	18.9

Experiment A is the baseline system, requiring a search effort of about 50000 hypotheses. The pruning threshold in the beam search was chosen such that there were less than two search errors per speaker. Eliminating all search errors would have required more than 100000 state hypotheses. In experiment B, the tree organization was used for the lexicon, and there is a drastic reduction in the number of state hypotheses by a factor of 10. However, a unigram rather than a bigram language model was used to avoid the need of having a separate tree copy for each of the predecessor words. Experiment C shows that the effect of making copies is not critical: the search effort is increased by roughly a factor of 2 over the unigram case. The search effort can significantly be reduced by using a look-ahead technique at the phoneme level as shown by experiment D. The figure for the search effort in this case includes the hypotheses for the tree search and the phoneme look-ahead. The overall improvement in the search cost is a reduction from 50000 to 3000 state hypotheses per 10-ms segment, i.e. a factor of about 17, while the average word error rate is not increased.

6. SUMMARY

This paper has presented techniques for improving time synchronous beam search for continuous speech recognition by using a tree organization of the pronunciation lexicon and a phoneme look-ahead technique. For a 10000-word task, the search effort was thus reduced by a factor of 17 without a loss in recognition accuracy. Ignoring the log-likelihood calculations of the emission probabilities, we obtained the result that the search can be performed in real time using a 100-MIPS processor.

References

[1] H. Ney, D. Mergel, A. Noll, A. Paeseler, *A data-driven organization of the dynamic programming beam search for continuous speech recognition*, Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Proc., Dallas, TX, pp. 833-836, April 1987.

[2] A. Paeseler, H. Ney, *Continuous-speech recognition using a stochastic language model*, Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Proc., Glasgow, pp. 719-722 , May 1989.

[3] H. Ney, *Experiments on mixture-density phoneme-modelling for the speaker-independent 1000-word speech recognition DARPA task*, Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Proc., Albuquerque, NM, pp. 713-716, April 1990.

[4] J. W. Klovstad, L. F. Mondshein, *The CASPERS linguistic analysis system*, IEEE Trans. on Acoustics, Speech and Signal Proc., Vol. ASSP-23, pp. 118-123, Feb.75.

[5] P. Laface, G. Micca, R. Pieraccini, *Recognition of Words in Very Large Vocabulary*, NATO ASI 1987, H. Niemann, M. Lang, G. Sagerer (eds.): 'Recent Advances in Speech Understanding and Dialog Systems', Springer, Berlin, pp. 235-254, 1988.

[6] L. Bahl, S.V. De Gennaro, P.S. Gopalakrishnan, R.L. Mercer, *A fast approximate acoustic match for large vocabulary speech recognition*, Proc. Europ. Conf. on Speech Communication and Technology, Paris, pp. 156-158, Sep. 1989.

[7] X. L. Aubert, A fast lexical selection strategy for large vocabulary continuous speech recognition, NATO ASI 1990, P. Laface (ed.): 'Speech Recognition and Understanding: Recent Advances, Trends and Applications', Springer, Berlin, to appear, 1991.

[8] H. Ney, *Acoustic modelling of phoneme units for continuous speech recognition*, Proc. Europ. Signal Processing Conf., Barcelona, pp. 65-72, Sep. 1990.

[9] R. Haeb-Umbach, H. Ney, *A look-ahead search technique for large vocabulary continuous speech recognition*, Proc. Europ. Conf. on Speech Communication and Technology, Genova, pp. 495-498, Sep. 1991.