

IMPROVEMENT IN N -BEST SEARCH FOR CONTINUOUS SPEECH RECOGNITION

Irina Illina and Yifan Gong

CRIN/CNRS, INRIA-Lorraine,
B.P. 239, 54506 Vandœuvre-lès-Nancy, France
{illina, gong}@loria.fr

ABSTRACT

In this paper, several techniques for reducing the search complexity of beam search for continuous speech recognition task are proposed. Six heuristic methods for pruning are described and the parameters of the pruning are adjusted to keep constant the word error rate while reducing the computational complexity and memory demand. The evaluation of the effect of each pruning method is performed in Mixture Stochastic Trajectory Model (MSTM). MSTM is a segment-based model using phonemes as the speech units. The set of tests in a speaker-dependent continuous speech recognition task shows that using the pruning methods, a substantial reduction of 67% of search effort is obtained in term of number of hypothesised phonemes during the search. All proposed techniques are independent of the acoustic models and therefore are applicable to other acoustic modeling techniques.

1. INTRODUCTION

In a continuous speech recognition task, the search for the best word sequence is one of the most time and memory consuming operations. This is because in time-aligning the input utterance with the reference acoustic models, a high number of possible paths have to be considered. A variety of approaches based on Viterbi search [8], N -best search, A* search [6] and stack decoding search [1] form the basis for most works in this domain. Some improvement of these methods have been proposed recently. A method for performing the optimal A* search, which guarantees to find the most likely path and with linear search time in the stack decoder is presented in [5]. Improvements in a time synchronous beam search strategy based on a tree-organisation of the pronunciation lexicon and the look-ahead technique are studied in [7] and [4].

In our work, we use a level-building N -best search. Using acoustic and language models, this method produces most likely N sentence hypotheses of utterance. A *sentence hypothesis* is a presumed partial sentence, a *phoneme symbol hypothesis* is a presumed phoneme symbol and a *level* is associated to a phoneme symbol in a sentence. As in other search techniques, the main difficulty of this search is the complexity. As most recognition tasks have very high number of hypotheses, a full search is impracticable on current computers. To achieve a good recognition rate, the value of N could be exponential in the length of the input vector sequence to be recognised.

For the reduction of the search space, an optimal use of all available knowledge in the pruning process is necessary. Such knowledge includes acoustic models, language model and the observed acoustic data.

In this paper, six heuristic pruning methods for N -best search are proposed. These methods are based on better use of available knowledge mentioned above. Experimental analysis of the reduction of search effort is given. The search effort is measured in term of average number of phoneme symbol hypotheses generated during the search per speaker and per recognised utterance.

The organisation of the paper is as follows. In section 2, the description of our baseline MSTM recognition system and the speaker-dependent continuous speech recognition task are presented. Pruning methods and its integration into the N -best search are described in section 3. The paper ends with a summary of the results and a conclusion.

2. RECOGNITION SYSTEM

2.1. Recognition system description

MSTM is a segment-based model using phonemes as the basic units. In MSTM, the observations of a phoneme are modeled by a set of stochastic trajectories. The trajectories are modeled by a mixture of probability density functions (pdf) of state sequences. Each state is associated with a multivariate Gaussian density function. A description of the details of acoustic-phonetic modeling can be found in [2]. The sentence recognition is based on the time function of probability for each phonetic symbol. The algorithm complexity for evaluating this function is the product of the number of time slots and the number of phonetic symbols. The language model used is based on [3] and is compiled into a network of phonetic symbols. The basic N -best sentence algorithm is as follows:

1. for each partial sentence hypothesis extend the sentence to all possible next phonemes using language model;
2. evaluate the score of the extended hypotheses;
3. sort the hypotheses;
4. prune the hypotheses.

2.2. Database description

Experiments deal with the French continuous speech corpus CEA recorded by the CRIN laboratory. For training, 79 sentences were read by 4 French speakers (males). In average, there are about 70 observations per phoneme for each speaker in the training part of the corpus. For testing, 241 sentences were recorded. There is only a small overlap between training and test vocabularies. The task is difficult because of insufficient training data and because of between word pauses which are not modeled by our grammar. Speech is sampled at 16 kHz. 13th order mel-cepstral vectors were computed every 10 ms with an analysis window of 32 ms. For this corpus, 32 context-independent phone models, including one silence model, are build. The language model has a word-pair equivalent perplexity of 48 and a 2010 words vocabulary. In all experiments, the covariance matrix is assumed to be diagonal and the number of Gaussian densities is about 600 per speaker.

3. PRUNING TECHNIQUES

In this section, six different heuristic pruning methods are presented. For each method, we give a description, a motivation and an evaluation of the search effort in term of the number of phoneme symbol hypotheses during the search, averaged over the set of recognised sentences and all speakers. A table of result gives the average number of phoneme hypotheses (PH), a percentage of number of word correctly recognised (%*Corr*), word accuracy (%*Acc*), number of word correctly recognised (Corr.) and the total number of deletion errors (D), substitution errors (S) and insertion errors (I). We call *search errors* the errors due to pruning.

In the first set of experiments, we studied the search size of the N -best search without using our new pruning technique (table 1). Varying the width of search (N) between 100 and 1300, the best word accuracy result 98.78% is achieved with a value $N = 1300$ and the search effort 89549 phoneme hypotheses.

N	PH	% Corr	%Acc	Corr	D,S,I
100	6044	94.48	93.40	5601	39,288,64
300	19119	97.6	97.15	5786	10,132,27
500	32665	98.29	97.99	5827	9,92,18
800	54031	98.66	98.36	5849	8,71,18
1000	68343	98.88	98.63	5862	4,62,15
1300	89549	99.05	98.78	5872	2,54,16

Table 1: Word accuracy rates as function of width of search (N).

In the second set of experiments given in the following, each pruning method is added to N -best search in the presented order and analyse of the reduction of search effort is performed. Linguistic information is used by pruning method 5. The other pruning methods use acoustic knowledge. The parameters of each pruning method are adjusted to avoid the search errors while reducing the computational complexity. The adjusted threshold is not too tight, because if it were the case, pruning errors would likely occur. The result for each method with an optimal choice of pruning threshold is given in following and summarised in section 4.

1. **Level dependent width of search.** During the search, the number of sentence hypotheses N is a function of the number of symbol levels recognised. In the N -best search, the important search effort is concentrated in the middle part of a sentence. For preserving an equal risk along whole search path, the width of search must be higher in the middle part than in the beginning and the end part of the search. During the search, the actual width of search is defined by:

$$N'(\ell) = N \cdot C(\ell) \quad (1)$$

where:

- N is the reference width of the search;
- $C(\ell)$ is the function of level ℓ , defined empirically.

Figure 1 represents the function $C(\ell)$.

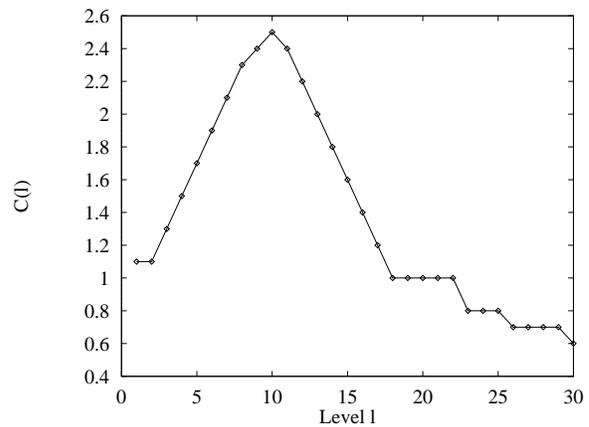


Figure 1: Function $C(\ell)$ for level dependent width of search

As one see in the line B of table 2, the important reduction of the search effort is obtained for the width of search of 500 ($N = 500$), keeping the 98.73% word accuracy. This number presents a reduction of 57% of the number of hypothesised phonemes (compared to the 89549 phoneme hypotheses for $N = 1300$ in table 1).

2. **Lower bound score.** Any sentence hypothesis whose score is lower than a given threshold (lower bound threshold, *LBT*) is pruned. This method is motivated by the fact that a low score of one sentence hypothesis has little chance to be significantly improved during the search. Line C of table 2 shows, that using the *LBT* equal to 0.48, the reduction of 6% of search effort can be achieved (compared to the 38077 phoneme hypotheses for $N = 500$ in line B of table 2).
3. **Score difference.** For each sentence hypothesis, the difference between the best score and the current score is calculated. A sentence hypothesis is pruned if this difference is larger than a given threshold (score difference threshold, *SDT*). As one can see in line D of table 2, the $SDT = 0.35$ leads to word accuracy (98.8%), generating 35782 phoneme hypotheses. This does not represent any search effort reduction (compared to the 35806 hypotheses in line C of table 2).

4. **Group score.** For each sentence hypothesis, the average score over M last symbols is calculated. A sentence hypothesis is pruned if this score is lower than a given threshold (group threshold, GT). Using $M = 9$ and $GT = 0.6$ (see line E of table 2), a search effort of 31369 phoneme symbol hypotheses is obtained, that presents 12% of reduction (compared to 35806 hypotheses).

5. **Modified A* search.** We propose to add at each node n of the network of phonetic symbols the number of longest and shortest paths from n to the final node of the network: $Min(n)$, $Max(n)$. During the recognition we estimate the number of remaining levels from the node m of the current level to the end of sentence. This estimation is based on the average speaking rate, which is calculated from the recognised part of the utterance. If the number of remaining levels is smaller than $Min(m)$ or larger than $Max(m)$, the sentence hypothesis passing through m is pruned. The motivation is that we want to eliminate the too short and too long hypothesised sentences with respect to length of sentence for recognition. The number of remaining levels is estimated as follows:

$$a(m) = \frac{(M - r(m)) \cdot \ell}{r(m)} \quad (2)$$

where:

- $a(m)$ is an estimation of the number of remaining levels for node m ;
- M is the length of the utterance (in frame);
- $r(m)$ is the length of the recognised part of the utterance (in frame) for node m ;
- ℓ is the current level.

Because of the low accuracy of the estimation of the number of remaining levels for the beginning part of the sentence, the following expression is used for pruning:

if $\ell \geq L$ and

$$a(m) \notin [Min(m)(1 - \varepsilon_1), Max(m)(1 + \varepsilon_2)]$$

then prune m

where $\varepsilon_1, \varepsilon_2$ are the parameters to be estimated and L is the minimal number of level for beginning the pruning. The best result is obtained with $L = 10$, $\varepsilon_1 = 0, 1$ and $\varepsilon_2 = 0, 5$ (EPS_1, EPS_2) (see line F of table 2). This represents the search effort of 29819 hypotheses (compared to line E of table 2) and the search effort reduction of 5%.

6. **Two pass search.** In the first pass, we perform an N -best search with a small width of search (2 or 3). The goal is to determine for the second pass a threshold, which is the score at the end of the first pass. During the second pass, a sentence hypothesis is pruned if its estimated score is smaller than this threshold. The experiments show that this method does not give any reduction of search effort because the found threshold is low and the sentence hypotheses corresponding to this threshold are pruned by other pruning methods (see line G of table 2).

4. SUMMARY OF THE RESULTS

This section presents the summary of the results, using the pruning techniques with optimal choice of parameters. Table 2 shows the search effort, percentage of reduction on search effort, adding one pruning method per one to N -best search.

	PH	%Corr	%Acc	Corr	D,S,I	R1	R2
A	89549	99.05	98.78	5872	2,54,16	0	0
B	38077	98.98	98.73	5868	3,57,15	57	57
C	35806	98.98	98.73	5868	3,57,15	6	60
D	35782	99.05	98.8	5874	1,55,15	0	60
E	31369	99.05	98.8	5872	1,55,15	12	65
F	29810	99.04	98.78	5871	1,56,15	5	67
G	29930	99.04	98.78	5871	1,56,15	0	67

Table 2: Summary of the search effort reduction, R1(%), R2(%). $N = 500$, $LBT = 0.48$, $SDT = 0.35$, $GT = 0.6$, $EPS_1 = 0.1$ and $EPS_2 = 0.5$.

We use $N = 500$, $LBT = 0.48$, $SDT = 0.35$, $GT = 0.6$, $EPS_1 = 0.1$ and $EPS_2 = 0.5$. The search effort reduction is given in percentage versus previous method (R1) and versus N -best search (R2). In this table, we use the following notation for recognition experiments:

- A: N -best search;
- B: N -best search and pruning method 1;
- C: N -best search and pruning methods 1 and 2;
- D: N -best search and pruning methods 1, 2 and 3;
- E: N -best search and pruning methods 1, 2, 3 and 4;
- F: N -best search and pruning methods 1, 2, 3, 4 and 5.
- G: N -best search and pruning methods 1, 2, 3, 4, 5 and 6.

The total reduction of 67% search effort can be obtained using the pruning techniques, without reducing the word accuracy and without the search errors. This presents the reduction of 57% of average execution time in a SUN SPARC 10/40 workstation (from 63 seconds per utterance to 27 seconds). The most significant reduction of the search effort is obtained with the level dependent width of search pruning method and with the group score method.

5. CONCLUSION

In this paper, six pruning techniques for improving the beam search are proposed. They are based on the use of available acoustic and linguistic information. Our objective is to minimize the computational cost with a minimal decrease in recognition accuracy. Experimental tests are reported on speaker dependent recognition task using Mixture Stochastic Trajectory Models. The study of the dependence of the search effort and the recognition accuracy as a function of the pruning parameters is presented. Experiments show that the methods 1, 2, 4 and 5 give a 67% total reduction of the search effort in term of number of hypothesised phonemes during the search.

In conclusion, for our recognition task, the 98.78% word accuracy with the 29810 hypothesised phonemes is obtained. All proposed techniques are independent of the acoustic models.

The use of pruning techniques is very important for continuous speech recognition related to large vocabulary, detailed acoustic modeling and a high language perplexity. This aspect will be investigated in our future work.

6. ACKNOWLEDGEMENT

The authors would like to express their appreciation to Jean Lieber and Olivier Siohan for fruitful discussions about this work.

7. REFERENCES

1. L. R. Bahl, F. Jelinek, and R. Mercer. A maximum likelihood approach to continuous speech recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-5(2):179–190, March 1983.
2. Y. Gong and J. P. Haton. Stochastic trajectory modeling for speech recognition. *In Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, 1:57–60, April 1994. Adelaide, Australia.
3. W. Katke. Learning language using a pattern recognition approach. *The AI Magazine*, 1985. Spring.
4. S. Ortmanms and H. Ney. Experimental analysis of the search space for 20 000-word speech recognition. *In Proc. of European Conference on Speech Communication and Technology*, 2:901–904, september 1995. Madrid, Spain.
5. D. B. Paul. Algorithms for an optimal A* search and linearizing the search in the stack decoder. *In Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, 1:693–696, 1991.
6. J. Pearl. *Heuristics – Intelligent Search Strategies for Computer Problem Solving*. Addison-Wesley Publishing Co., Reading, MA, 1984.
7. V. Steinbiss, B.-H. Tran, and H. Ney. Improvements in beam search. *In Proc. of Int. Conf on Spoken Language Processing*, 3:2143–2146, March 1994.
8. G. M. White. Dynamic programming, the Viterbi algorithm, and low cost speech recognition. *In Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, 1:413–417, 1978.