COMBINING LENGTH RESTRICTIONS AND N-BEST TECHNIQUES IN MULTIPLE-PASS SEARCH STRATEGIES

José A. R. Fonollosa and Eloi Batlle
TALP. Dept. of Signal Theory and Communications. Universitat Politècnica de Catalunya.
c/ Jordi Girona 1-3. Edifici D5. Barcelona 08034, SPAIN
e-mail: adrian@gps.tsc.upc.es

ABSTRACT

Multiple-pass search strategies are a necessary choice to reduce the computational cost and memory requirements of large vocabulary speech recognition systems uisng different kinds or complex modelling of speech and language. In the N-best approach the most efficient knowledge sources (usually acoustic models and a bi/trigram language model) are used first to select a short list (N-best) of alternative hypotheses. Then, the remaining more complex knowledge sources are used to rescore the sequences. In this paper we propose a new algorithm to overcome the problems associated with the simple "Traceback-Based" N-best algorithm while conserving its simplicity and low computational cost.

1 INTRODUCTION

This paper presents a new N-best search algorithm based on the combination of two well-known techniques in continuous speech recognition: grammars that impose a restriction in the length (number of words) of the recognized sequence [3] and the standard traceback-based N-best algorithm [4, 5, 7]

Multiple-pass search strategies are a common choice to reduce the computational cost and memory requirements of large-vocabulary speech recognition systems [7]. In the N-best approach the most efficient knowledge sources (usually acoustic models and a bi/trigram language model) are used first to select a short list (N-best) of alternative hypotheses. Then, the remaining more complex knowledge sources (long-distance relations between words, language understanding models, valid telephone or credit card numbers, prosody, ...) are used to rescore and reorder the N-best hypotheses with a reasonable computational effort.

Even if we assume different hypotheses have significantly different scores, the exact N-best algorithm is of little practical interest because of its computational cost. Therefore, several approximate algorithms have been developed. The simplest and least expensive algorithm for finding the N-best hypotheses is one that simply uses the traceback information of the standard one-pass algorithm [1, 2, 6]. The traceback-based N-best algorithm requires almost no extra computation respect to the standard one-pass algorithm. However, in selecting the N-best hypotheses it usually misses or underestimates high scored ones.

The traceback-based N-best algorithm only keeps one path within a word and multiple hypotheses are considered only in the transitions between words. Hypotheses of different length as "one two three" and "one three" are never simultaneously consider by the traceback-based N-best algorithm even if both have almost the same probability and we consider a large number of hypotheses. Several approaches of compromise between the exact N-best and the traceback-based algorithm have been proposed in the literature to solve that problem. However the proposed algorithms usually represent a significant increase in the computational cost or memory requirements respect to the basic algorithm.

The approach proposed in this paper combines the traceback-based N-best algorithm with the length restriction imposed by a simple grammar to force the inclusion of sequences of different length in the selected list of word sequences. These kind of grammars are usual in tasks as connected digit recognition in which the expected number of digits may be forced by a finite-state grammar. Since multiple insertions or deletions are not probable, the grammar is usually simplified to consider sequences of length $Kn+m$ for any n, where K has a small fixed value and m ranges from 0 to $K-1$.

In the results presented here we have considered values of K in the range from 1 to 5, in combination with the traceback-based N-best algorithm with up to 200 hypotheses per sentence.

The following section give more details about the proposed algorithm. Then, we present the results obtained in two different tasks: the recognition of connected digits and the recognition of numbers.

[0]This research was supported by the CICYT Spanish research project TIC98-0683

2 EXTENDED TRACEBACK-BASED N-BEST ALGORITHM

The simplest and least expensive algorithm for obtaining multiple hypotheses was first proposed in [4, 5] as a modification of the one-pass algorithm. As in the standard one-pass algorithm, in this N-best algorithm we only keep one theory at each word state. However, at each grammar node, the traceback information is extended to store not only the index and score of the best scoring word, but also the indexes and scores of the n best-scoring words. At the end of the sequence, the saved traceback information is used to recursively generate the N best answers.

This modification supposes a negligible increase in the computational cost above the standard one-pass algorithm. However, this simplicity is also the cause of one important drawback. Since this traceback-based algorithm keeps only one theory a each state, it systematically misses some high scoring hypotheses.

For example, let us consider a simple task as connected digit recognition. If the best hypothesis is "one three", the standard traceback-based algorithm will not be able to generate other hypotheses beginning with the word "one" and ending with the word "three". That is, even if the sentence "one two three" has almost the same probability, it will never appear in the list of the n best generated hypotheses. This is because we keep only one copy of each word state. There is only one best beginning time for the ending word "three", and only one best ending time for the first word "one".

In the proposed extension of the standard algorithm, this problem is solved with the addition of a grammar that forces the consideration of sequences of different length in the N-best search. Figure 1 shows the 3-state grammar that considers sequences of $3n+i$ words at each node i, for any arbitrary n.

In general, the proposed N-Best algorithm adds a grammar of K states to the traceback-based N-best algorithm for connected word recognition. This simple grammar forces the consideration of different lists of hypotheses at each of the K nodes. In the implementation of the proposed algorithm considered here, the extended N-Best algorithm will keep a separate list with the N-best word sequences of length $nK+i$ at each grammar node i . At the end of the sequence, we will select the only N-Best global hypotheses among the total $K*N$ hypotheses stored in all the grammar nodes.

The proposed algorithm multiplies by K the search space, since it keeps a different word state theory at each grammar node. However, in practice, the proposed algorithm does not usually represent a significant increase in the total computational cost if we include a pruning technique as the beam search in our search.

If the following section we show how the proposed extended N-best algorithm may be useful in some recognition tasks. We will also include results comparing the performance and computational cost of the proposed algorithm with respect to the standard ($K = 1$) N-best algorithm for connected word recognition.

3 RESULTS

Several recognition experiments were conducted in order to evaluate the performance of the proposed N-best algorithm. In this paper we present the results obtained in two simple tasks: connected digit and number recognition..

The parameters that defines the common sections of the recognizer are the following:

- Sampling Frequency: 8000 samples/s
- Window size: 30 ms.
- Window displacement: 10 ms.
- Parameterization: 14 MFCC coefficients with cepstral liftering and 20 bank filters. Real-time cepstral mean substraction.
- Delta frames: 2
- Acceleration frames: 1

3.1 Task 1. Connected digit recognition

In this first experiment, we consider the recognition of digit strings using whole word models for the ten Spanish digits plus a model of the silence/noise. To train the HMM models of the digits we use part of the SpeechDat [9] Spanish Fixed Network Corpus, while for testing we considered a different database with 1200 digit strings of different lengths.

Here, we studied the cumulative percent of correct strings as a function of the number N of hypotheses and the number K of grammar states. The results are of direct interest for applications where the list of valid digit strings is restricted to a finite set, or the strings include some kind of error-detecting or error-correcting codification.

Figure 2 shows the percent of correct strings that were missing in the generated N-best sequences for different number of grammar nodes, $K = 1$, $K = 2$ and $K = 5$. The results for $K = 2$ and $K = 5$ shows a clear improvement with respect to the standard traceback-based N-best algorithm ($K = 1$) if we consider a large number of hypotheses. In this experiment, we have 11 words models (10 digits + noise) in parallel in each node of the grammar. Since we consider the model of the silence in parallel with the models of the digits, there is no direct relation between the grammar node and the number of digits in the hyphoteses, i.e., we can have sequence with zero, one or more silence 'words'. In fact, most of the hyphoteses only differ in the number and/or position of the silence 'words'.

Figure 3 compares the number of seconds required to process the test database, as a function of the number of grammar states and the number of hypotheses.

For $K = 2$, the total computational cost is similar to the cost of the standard algorithm ($K = 1$).

3.2 Task 2. Recognition of numbers

In this experiment, we studied the recogniton of large currency money amounts (greater than one million). As baseline we considered a simple connected word recognition system without any kind of grammar (number generation grammar or n-gram models). The vocabulary consisted of 70 words (uno, dos, tres, veintiuno, veintidos, treinta, treinta y, cien, doscientos, mil, un millón, pesetas, ...) created from demiphone models [8]. The demiphone models were trained using part of the phonetically-rich sentences of the SpeechDat database. The 625 numbers of test were selected from the same database.

For this task, we studied if the grammar-free recognition system was able to provide the correct answer, and if the correct answer was the first hypothesis grammatically correct. This grammar-free system was intentionally too simple to provide results of practical interest in any case. The purpose of the experiment was just to study the behaviour of the different algorithms.

The results of the baseline system ($N = 1, K = 1$) as well as the results obtained with the standard and the extended traceback-based algorithm N-best are presented in table 1. We considered that the recognizer was providing a correct answer when the first hyphotesis corresponding to a single grammatically-correct number was correct. If this first number was not correct we had an error. If the recognizer did not provided any grammatically-correct number, we considered the sentence as *rejected.*

N	K	*correct*	*errors*	*rejected*
1	1	260	74	291
10	1	340	124	161
10	2	332	109	184
100	1	360	187	78
100	2	368	166	91

Table 1. Comparison of the results obtained by the baseline system ($N = 1, K = 1$), the standard traceback-based N_Best algorithm ($N > 1, K = 1$), and the proposed algorithm ($K = 2$). A number is *rejected* if the recognizer is does not generate a grammatically-correct hypothesis.

The results indicates that the proposed N-Best algorithm with $K = 2$ reduces the number of incorrect recognized number from 124 ($K = 1$) to 109, when we consider the first 10 hypotheses, ($N = 10$). We can also observe that increasing the number of hyphoteses from 10 to 100 produced much more wrong hypotheses

4 CONCLUSIONS

This paper studied the combination of two well-known techniques in speech recognition: grammars that impose a restriction in the length (number of words) of the recognized sequence and the traceback-based N-best algorithm. The proposed algorithm alleviates the problems of the standard traceback-based N-best algorithm, while keeping its simplicity.

The results in tasks as recognition of digit strings and numbers from the Spanish SpeechDat Database [9] indicates that the proposed algorithm is a good choice for real-time applications. The algorithm has been already selected for commercially available speech recognition systems and applications.

References

[1] T.K. Vintsyuk. "Element-Wise Recognition of Continuous Speech Consisting of Words From a Specified Vocabulary," Kibernetica (Cybernetics), No. 2, pp. 133-143, March-April 1971.

[2] H. Ney. "The Use of a One-Stage Dynamic Programming Algorithm for Connected Word Recognition," IEEE Trans. Acoustics, Speech, Signal Proc., Vol. 32, no. 2, pp 263-271, April 1984.

[3] C.H. Lee and L. Rabiner, "A Frame-Synchronous Network Search Algorithm for Connected-Word Recognition," IEEE Trans. Acoustics, Speech, Signal Proc., Vol. 37, no. 11, pp 1649-1658, November 1989.

[4] J.B. Mariño and E. Monte. "Generation of Multiple Hypothesis in Connected Phonetic-Unit Recognition by a Modified One-Stage Dynamic Programming Algorithm," Proc. Eurospeech 89, Vol. 2, pp.408-411. Paris, September 1989.

[5] V. Steinbiss. "Sentence-Hypotheses Generation in a Continuous-Speech Recognition System," Proc. Eurospeech 89, Vol. 2, pp.51-54. Paris, September. 1989.

[6] Lawrence Rabiner, Biing-Hwang Juang. *Fundamentals of Speech Recognition.* Prentice Hall 1993.

[7] R. Schwartz, L. Nguyen and J. Makhoul. "Multiple-Pass Search Strategies". In Automatic Speech and Speaker Recognition, Kluwer Academic1996. Chapter 18.

[8] José B. Mariño, A. Noguerias and A. Bonafonte. "The demiphone: an efficient subword unit for continuous speech recognition," Proc. EUROSPEECH'97, pp. 1215-1218, Rhodes, September 1997.

[9] SpeechDat II. Speech Databases for the creation of voice driven teleservices. Language Engineering Resources. EC Telematics. http://www.speechdat.org.

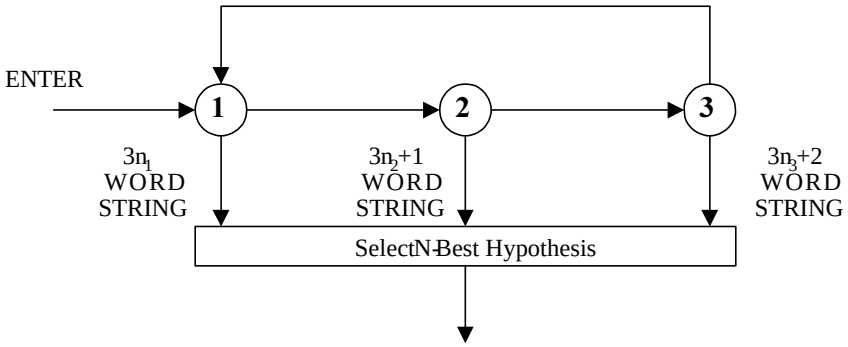

Figure 1: Grammar network generating connected word strings of different length

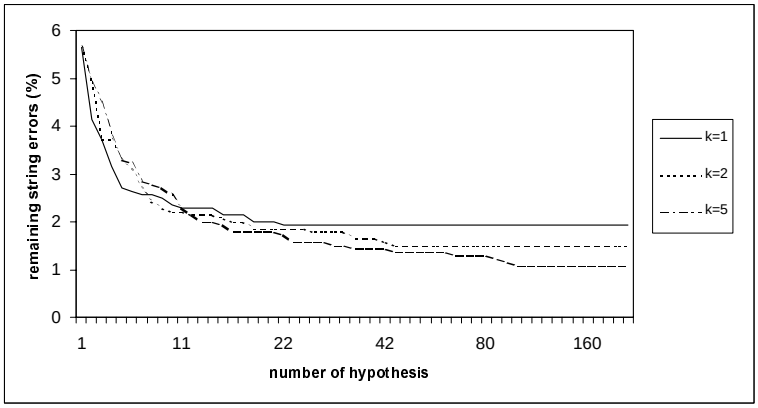

Figure 2: Comparison of the percent of missed correct digit strings as a function of the number of grammar nodes k and the number of computed hypotheses.

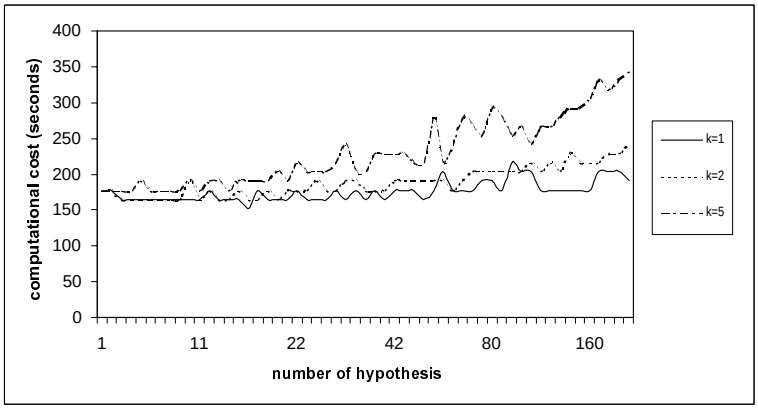

Figure 3: Comparison of the computational cost of the proposed algorithm as a function of the number of grammar nodes k and the number of computed hypotheses (Task 1).