MINIMIZING SEARCH ERRORS DUE TO DELAYED BIGRAMS IN REAL-TIME SPEECH RECOGNITION SYSTEMS

M.Woszczyna *M.Finke*

INTERACTIVE SYSTEMS LABORATORIES
at Carnegie Mellon University, USA
and University of Karlsruhe, Germany

ABSTRACT

When building applications from large vocabulary speech recognition systems, a certain amount of search errors due to pruning often has to be accepted in order to obtain the required speed. In this paper we tackle the problems resulting from aggressive pruning strategies as typically applied in large vocabulary systems to achieve close to real-time performance. We consider a typical scenario of a two pass viterbi search with the first pass being organized as a phoneme (allophone) tree. For such a tree organized lexicon, there are two possiblities to use a bigram language model: either by building tree copies or by using so-called delayed bigrams. Since copying trees turns out to be too expensive for real time applications we basically refer to delayed bigrams, discuss their drastic influence on the word accuracy and show how to alleviate the desastrous effect of delayed bigrams under aggressive pruning.

1. INTRODUCTION

Many approaches used for large vocabulary speech recognition require a time synchronous viterbi search as first pass, which is either used as a lookahead for an A^* search or to restrict the search space for a more detailed viterbi search. Since a large number of words in the vocabulary begin with the same initial sequence of phonemes or allophones, it is advantageous to arrange the pronunciation lexicon as a tree. Each node in the tree stands for an allophone such that a path from the tree root to a tree leaf represents a legal allophone sequence and thus a legal word in the vocabulary.

Compared to a linear (flat) organisation of the vocabulary the tree structure causes a problem when including language models at word transitions: expanding from the end of a word w_1 to the beginning of the next word is done by expanding into the tree root. But when a tree is started, all words are hypothesized and the word identities are only known at the end of the tree. Therefore, the transition probability $p(w_2|w_1)$ which is typically a bigram language model score cannot be computed immediately upon transition. There are two solutions to this problem: either tree copies are generated for each active word end at a given frame [2] or the bigram score is not added before reaching the leaf of the tree and thus the word identity is known (delayed bigram approach). Since creating tree copies is often too expensive for a fast first pass of a multipass search, we focus on the benefits and problems of using delayed bigrams instead.

In this paper we investigate the effects of using delayed bigrams in combination with real-time performance oriented and thus kind of aggressive pruning conditions on our JANUS speech recognition demo system [1]. Simulations demonstrate the often desastrous effect of the delayed language model approach under these special circumstances. We also study different strategies of recovering from these additional search errors caused by using delayed bigrams.

The experiments presented in this paper are performed on two different tasks; The first set of test data is composed of 102 german sentences chosen randomly from utterances recorded with our demo system. For testing a 3500 word vocabulary and a bigram language model are used. In the demonstration the subjects speak to other people via a computer. The resulting sentences are inherently shorter and easier to recognize than sentences collected in fully human-to-human dialog setup usually used for collecting data for the German Spontaneous Scheduling Task (GSST). However, it seemed to be of more practical relevance to examine the effects of pruning on a typical on-line demo situation than on a typical off-line evaluation system where word accuracy losses are often not acceptable.

The second set are the first 10 minutes of speech form the 1994 WSJ evaluation with a vocabulary 20000 words and a trigram language model. These experiments are to verify that the conclusions derived form the experiments on spontaneous speech with medium vocabulary size and bigrams still hold for this completely different application.

2. DELAYED BIGRAMS

In a linear as well as in a tree organized vocabulary delayed bigrams have two main advantages compared to standard (immediate) bigram language models:

- as they are added before entering the last phoneme of a word (which for a tree organized vocabulary is a tree leaf) they can be used even when the vocabulary is organized as a tree without the necessity of creating tree copies.
- most word hypotheses are pruned away before they reach the end of the word. Delayed bigrams only have to be computed for the remaining word-ends. Thus, the total amount of language model queries can be reduced by a factor 10 to 20.

These benefits have to be paid by two kinds of search errors, those which are inherent in the algorithm and independent

of the beam size, and those who get worse when the beams are reduced to build real time systems.

2.1. Beam independent search errors

When a path is expanded into a tree root, the best matching acoustic word end w_1 is stored as the predecessor in the new path (backtrace). Later, when this path is expanded to a tree leaf from the penultimate into the last phoneme, the bigram score is computed and the backpointer adjusted as follows: at this point the identity of the current word w_2 is known. All words ending at the frame where w_2 started are considered possible predecessor candidates of w_2 and the candidate with the lowest total score (the accumulated score up to the end of the candidate plus the bigram penalty into the current word) becomes the predecessor of w_2.

However, the information about where w_2 started is not modified. This assumes that the ideal starting point of a word is independent of the identity of the predecessor word. The problem is, that a predecessor word which is expanded into the tree root at a different point of time might loose against the locally best path even though its total score after adding the language model would be better.

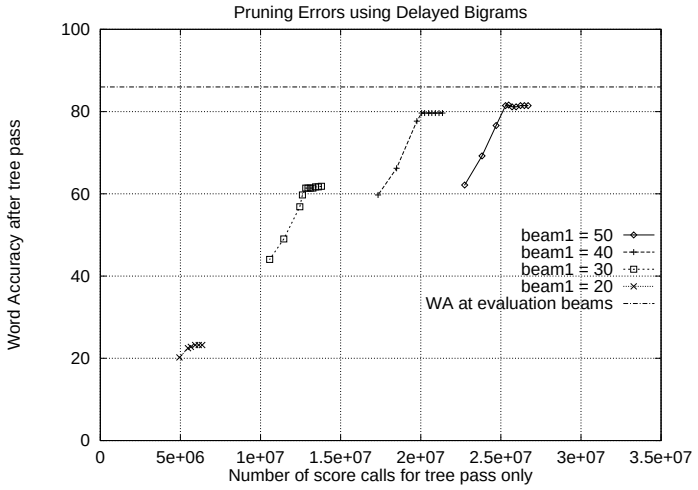
Obviously, there is no way to recover from this kind of search errors by choosing a larger beamwidth. We have to add a second linearly organized pass to the algorithm instead. Because of these beam independent search errors the JANUS recognition engine uses the tree pass to select likely starting points for words only and then does a second flat pass using standard bigram models.

2.2. Beam dependent search errors

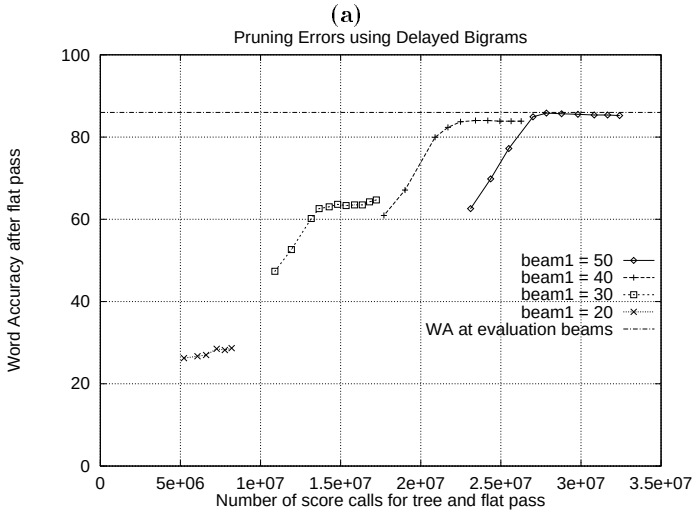
Figure 1 demonstrates how for reasonable large beam sizes, nearly the whole search error due to using delayed bigrams in a tree can be recovered by a second path. The four curves represent four different settings of the main beam used to prune the nodes within the tree. The data points on each of these curves represent different settings of the secondary beam that is used to prune the competing tree leafs only[1]. The word accuracy of the recognizer is plotted over the number of calls to the score routine, which can be used as a machine independent measure of the volume of the search space remaining after pruning.

Figure 1 also reveals that for smaller beams the recognition performance is far from degrading gracefully. On the one hand, even if a 5% word accuracy loss due to pruning were acceptable, the number of required score computations could only be reduced by about 25%. On the other hand, to get a faster recognition engine (e.g. to achieve real-time performance) you have to reduce the beams to such an extend that virtually no recognition performance can be achieved. The reason for this behavior is that the bigram information is added later for a delayed bigram than for a standard bigram. Therefore, words that do not match well acoustically but would get a good bigram score *later* are likely to be pruned away before they reach their last phoneme.

[1] The second leaf related beam was introduced to control the number of language model requests (when entering the leaf node) and word transitions (i.e. expanding the leaf node to the tree root(s)) individually.



(a)



(b)

Figure 1. **(a)** Search errors due to tight pruning in tree pass. **(b)** For small beams the pruning errors due to delayed bigrams in the first pass cannot be recovered by the second linear pass. But for a beam > 50 it is possible to achieve the original evaluation beam performance with the flat pass corrected output again (see 2.1).

3. MINIMUM UNIGRAM LOOKAHEAD

In order to compensate the effect described above the idea is to get an estimate of how well a branch of the tree will do including language model information as early as possible. We tried to use the following **minimum unigram approximation**:

> For each node in the tree, the minimum unigram penalty for all words in the subtree is computed. This approximation is more accurate for nodes that are close to the tree leafs, less accurate for nodes that are close to the root. At each phoneme transition the inaccurate estimate of the node before is subtracted from the total score and replaced by the more accurate estimate of the next node.

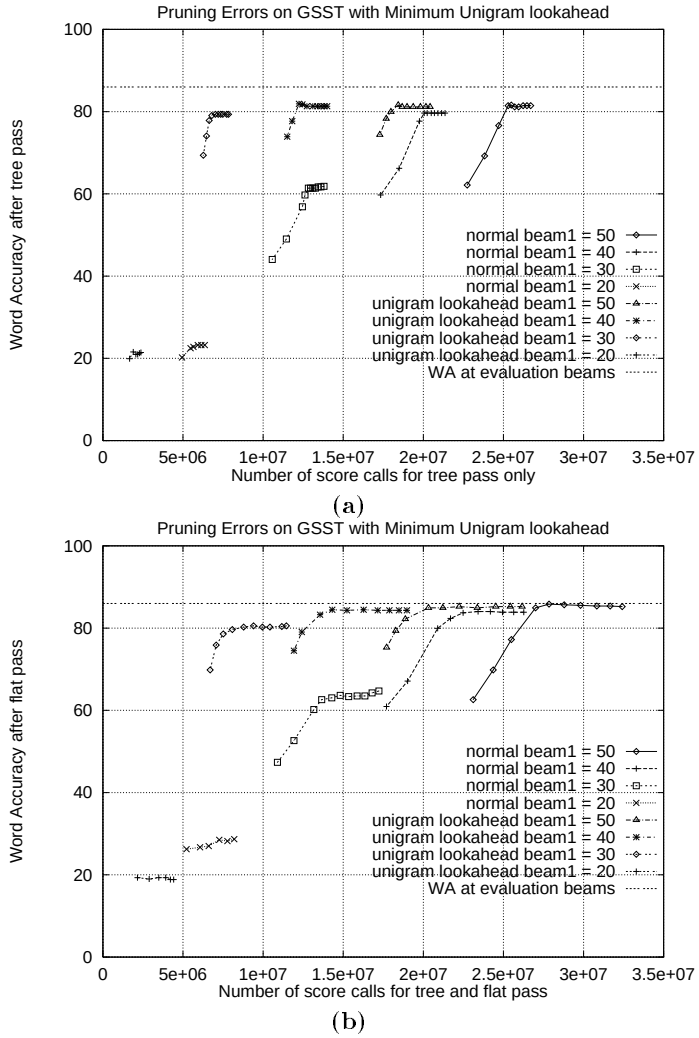


Figure 2. **(a)** Pruning errors are reduced due to minimum unigram lookahead on GSST. **(b)** Error reduction also helps for second pass.

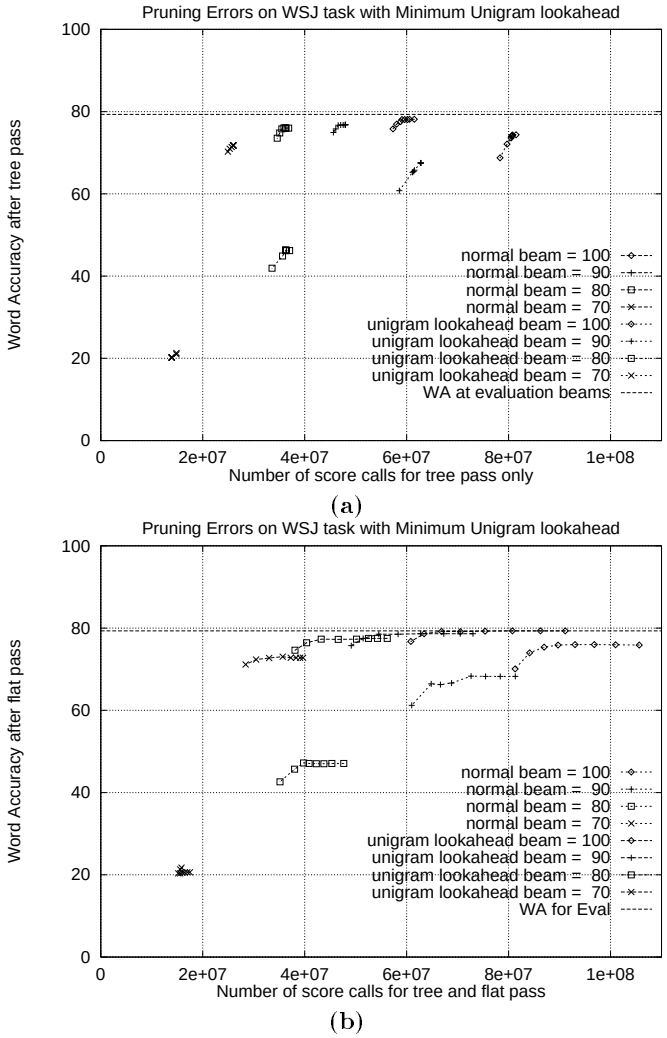


Figure 3. **(a)** Pruning error reduction with minimum unigram lookahead on WSJ. **(b)** Result after second pass.

Figure 2 shows that using the proposed language model lookaheads within the tree pass the word accuracy remains very stable over a large range of beams. With a word accuracy loss of about 5% a speedup by 65% can be achieved. Only at very small beams the word accuracy drops drastically to 20%.

Figure 3 shows that the same algorithm also helps to avoid pruning errors in a demonstration system for the 20000 word Wall Street Journal dictation task. The WSJ test were run on the first 10 minutes of the official 1994 evaluation set.

4. MINIMUM BIGRAM LOOKAHEAD

For the plots in figure 4 we refer to a slightly modified lookahead technique. Instead of considering the minimal unigram penalty as lookahead score we used minimal bigram scores where for each word w_i we selected the minimal bigram penalty $\min_{w_j} p(w_i|w_j)$. It turns out that this kind of lookahead performs better than using no lookahead at all but slightly worse than the minimum unigram lookahead. Part of the problem of this approach is that, close to the root of tree, the lookahead score is always close to 0 which is comparable to the situation of having no lookahead at all.

5. CONCLUSIONS

In this paper we demonstrated that there seems to be a very poor degradation behavior in a speech recognition engine given its first pass is tree organized and based on delayed bigrams as language model. We observed a drastic influence of the delayed bigram approach on the word accuracy in a setting where aggressive pruning has to be used to achieve close to real-time performance. In order to alleviate the desastrous effect of delayed bigrams under these circumstances we proposed and evaluated a new kind of language model lookahead technique which makes a speech recognition engine much more robust against search errors due to pruning.

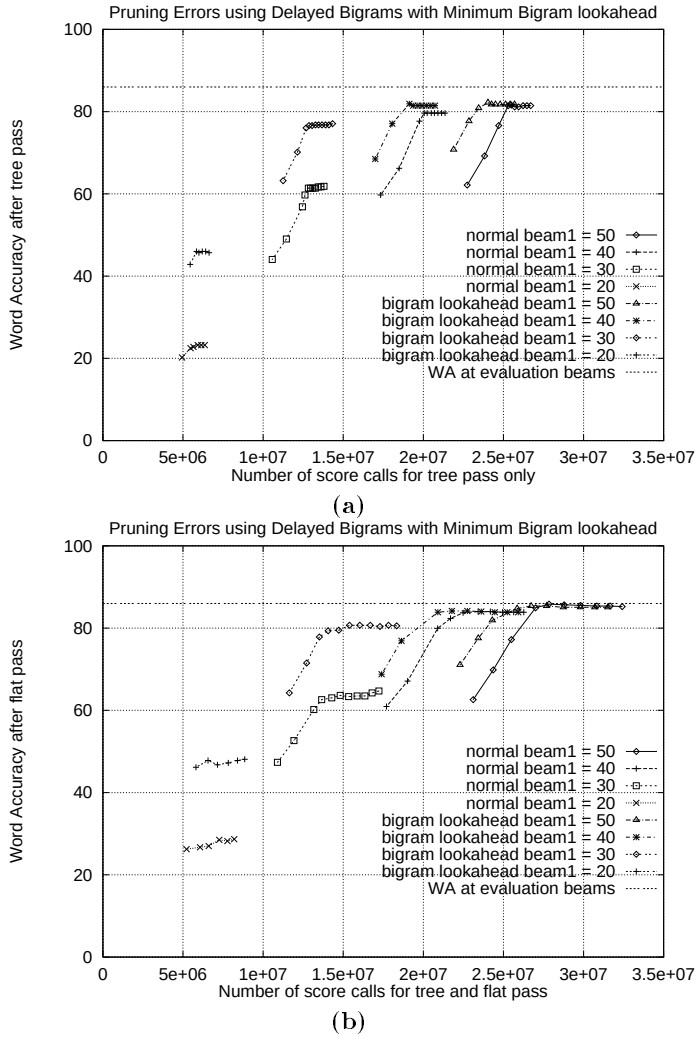


Figure 4. **(a)** Pruning errors are reduced due to bigram lookahead. **(b)** Error reduction also helps for second pass.

6. ACKNOWLEDGEMENTS

Many thanks to Fil Alleva for helpful discussion and valuable insights on using delayed bigrams.
This work was funded in part by grand 413-4001-01IV101S3 from the German Federal Ministry of Education, Science, Research and Technology (BMBF) as part of the VERBMOBIL project.

REFERENCES

[1] A.Waibel, M.Finke, D.Gates, M.Gavaldà, T.Kemp, A.Lavie, L.Levin, M.Maier, L.Mayfield, A.McNair, I.Rogina, K.Shima, T.Sloboda, M.Woszczyna, T.Zeppenfeld, P.Zhan *JANUS-II — Advances in Spontaneous Speech Translation* ICASSP96;

[2] V.Steinbiss,B.H. Tran, H.Ney *Improvements in Beam Search* ICSLP'94 Volume 4 pp 2143–2147;

[3] X.Aubert, H.Ney *Large Vocabulary Continuous Speech Recognition Using Word Graphs* ICASSP'95, Volume 1 pp 49–52;

MINIMIZING SEARCH ERRORS DUE TO DELAYED BIGRAMS IN REAL-TIME SPEECH RECOGNITION SYSTEMS

M.Woszczyna and M.Finke
INTERACTIVE SYSTEMS LABORATORIES
at Carnegie Mellon University, USA
and University of Karlsruhe, Germany

When building applications from large vocabulary speech recognition systems, a certain amount of search errors due to pruning often has to be accepted in order to obtain the required speed. In this paper we tackle the problems resulting from aggressive pruning strategies as typically applied in large vocabulary systems to achieve close to real-time performance. We consider a typical scenario of a two pass viterbi search with the first pass being organized as a phoneme (allophone) tree. For such a tree organized lexicon, there are two possiblities to use a bigram language model: either by building tree copies or by using so-called delayed bigrams. Since copying trees turns out to be too expensive for real time applications we basically refer to delayed bigrams, discuss their drastic influence on the word accuracy and show how to alleviate the desastrous effect of delayed bigrams under aggressive pruning.