

# Lattice Compression in the Consensual Post-Processing Framework

Lidia Mangu and Eric Brill  
Computer Science Dept., Johns Hopkins University  
Baltimore, MD 21218, USA

## ABSTRACT

Word Lattices are used by most speech recognizers as a compact representation of a set of alternative hypotheses. In large-vocabulary, multi-pass recognition systems it is important to generate word lattices incorporating a large number of hypotheses but at the same time keeping the size of the representation as small as possible. Previously we presented a method for identifying mutually supporting and competing word hypotheses in a recognition lattice. In this paper we show how the outcome of this method can be used for compressing lattices. The success of the new technique comes from the ability to discard links with low a posteriori probability and recombine the remaining ones to create a new set of hypotheses. Experiments on the Switchboard corpus show that this method results in better compression results than the conventionally used technique.

**Keywords:** Large Vocabulary Speech Recognition, Lattice Compression, Lattice Decoding, Switchboard

## 1 INTRODUCTION

Word Lattices are used by most speech recognizers as a compact representation of a set of alternative hypotheses. A lattice is a graph in which each link is a word with time information and the likelihood that the word was uttered in that particular time interval (see Input in Figure 1). In large vocabulary, multi-pass recognition systems it is important to have small but still highly accurate lattices. The computational cost of search algorithms that incrementally make use of more detailed acoustic models and better language models is correlated with the size of the lattices that constrain the search. It is desirable to have lattices with a large number of paths so as to minimize the search errors. However, large lattices make subsequent lattice post-processing steps or recognition passes slow. In the standard beam-pruning method [8] the likelihood of the most likely path through each link is compared with the likelihood of the most likely path through the lattice; all

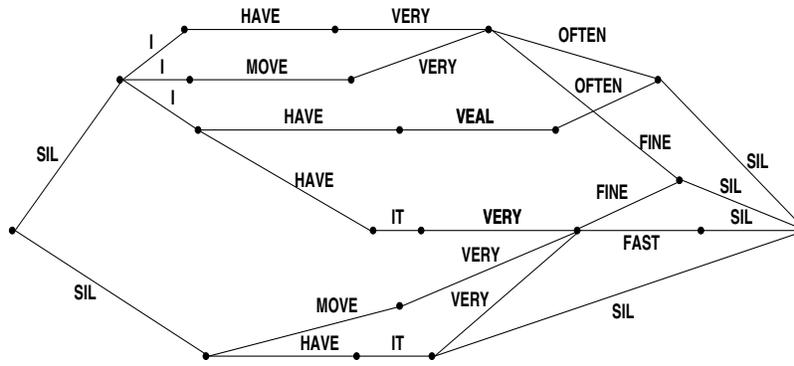
the links for which the difference between these two likelihoods falls outside the beam width are removed from the lattice. Any nodes disconnected as a result of the previous step are also removed. Thus this method reduces the size of the lattice (possibly reducing its coverage) by removing the least likely hypotheses.

In [4] we introduced a method for extracting word hypotheses with the highest posterior probabilities from word lattices and showed that this led to a significant reduction in WER over the standard MAP decoding approach [1]. The core of this method is a clustering procedure that identifies mutually supporting and competing word hypotheses in a lattice. This can be viewed as a process of converting the lattice into a word graph with a different topology which we refer to as a *confusion network*. This paper shows how this new representation can be used for pruning lattices and compares its effectiveness against the traditional likelihood-based method. Effectiveness on this task is judged in terms of lattice accuracy and size of representation. Experiments on the Switchboard LVCSR task show that our method results in significantly smaller representations at a given lattice accuracy level. We also show that by running a new recognition pass which takes advantage of the newly hypothesized strings of words we obtain lattices having the same accuracy as the original ones while being three times smaller. The rest of the paper is organized as follows. In Section 2 we describe the method for obtaining the confusion networks. Then in Section 3 and 4 we describe two methods for compressing lattices based on the result of Section 1. Conclusions are given in Section 5.

## 2 CONSENSUAL POST-PROCESSING

In the standard MAP approach [1] a speech recognition system aims to find the word sequence  $W$  that has the highest posterior probability  $P(W|A)$  for a given acoustic waveform  $A$ . The commonly used performance metric is the recognition *word error rate* (WER) defined as the percentage of incorrectly recognized words. In [10] it is shown that the mismatch between the standard MAP paradigm which is sentence-based and the standard evaluation metric which is word-based, can lead to sub-optimal recognition results. In [4] we described a method for ad-

### Input Lattice:



### Output Confusion Graph:

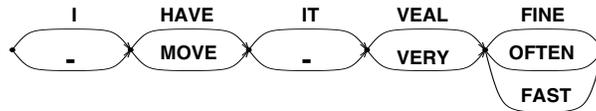


Figure 1: From Lattices to Confusion Networks

addressing this discrepancy namely explicitly minimizing the word error metric by extracting high posterior probability word hypotheses from word lattices. In order to find the word hypotheses with the highest posterior probabilities we have to find a complete alignment of all words in the lattice, identifying mutually supportive and competing word hypotheses. The difficulty of this task comes from the fact that lattices only impose a partial temporal order on word hypotheses. We therefore developed a clustering algorithm that groups word hypotheses into time-synchronous slots based on both their temporal and phonetic similarity.

Formally, an alignment consists of an equivalence relation over the word hypotheses together with a total ordering of the equivalence classes, such that the ordering is consistent with that of the original lattice. Our method for obtaining the alignment is to first induce a partial order on sets of links based on the precedence order on the graph, and then cluster sets of links so as to derive a total order. The clustering algorithm has two steps: (1) Intra-word clustering and (2) Inter-word clustering. During step (1) we try to group together links which overlap in time and correspond to the same word. In this case the similarity between two clusters is computed based on the degree of overlap between the time intervals in which the words were hypothesized, weighted by the link posterior probability. The posterior probability of a link is the sum of the posterior probabilities of all the paths the link is part of. The temporal overlap is weighted by the link posterior probability so as to make the measure less sensitive to unlikely word hypotheses. At step (2) we start grouping together clusters corresponding to different words based

on the word phonetic similarity between them, weighted again by the posterior probability of the link components. At each step, the most similar clusters are merged if there is no precedence relation between them. The merging process is repeated until there are no more candidates, which is equivalent to having a total order on the set of clusters. If we start with a partial order and at each step we merge clusters which are not in relation then we are guaranteed to generate a total order in a finite number of steps.

In Figure 1, we see many links corresponding to the word **HAVE**, hypothesized in similar time intervals. The goal of the first step is to put together all such links. At the end of step (1) we will have a cluster corresponding to **HAVE**, one corresponding to **MOVE**, etc. In step (2) we would like to merge the clusters corresponding to **MOVE** and **HAVE** because they seem to have been hypothesized for the same word in the reference transcription: they are similar sounding words, they have time overlap and there is no path in the lattice connecting them.

The total posterior probability of an alignment class can be strictly less than 1. That happens when there are paths in the original lattice that do not contain a word at that position; the missing probability mass corresponds to the probability of a deletion (or null word). We explicitly represent deletions by a link “-”. For example, in the lattice in Figure 1 there are some hypotheses having “I” as the first word, while others have no corresponding word in that position. The final alignment thus contains two competing hypotheses in the first position: the word “I” (with posterior equal to the sum of all hypotheses starting with that word), and the null word (with posterior equal to the sum of all other hypotheses).

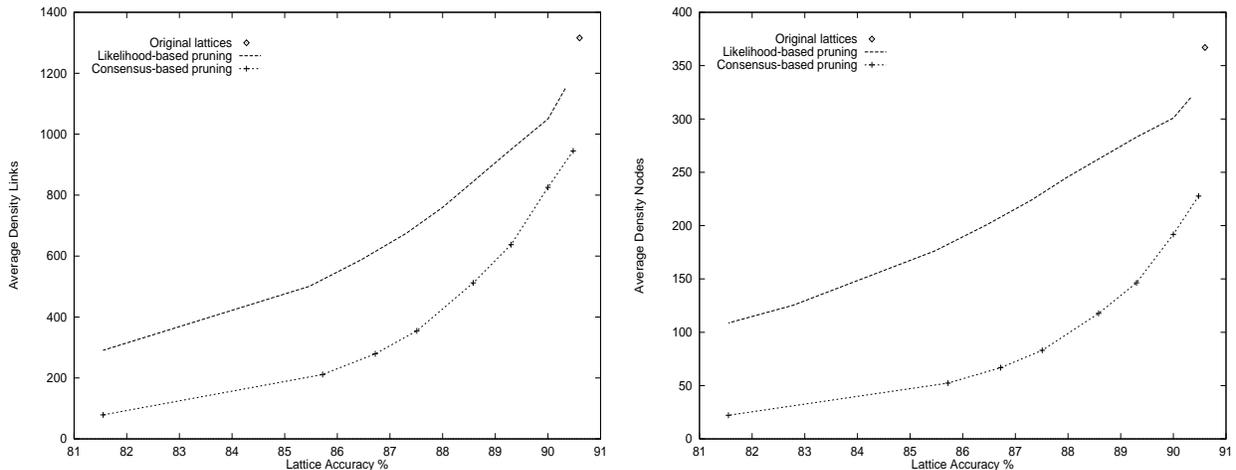


Figure 2: Effectiveness comparison between likelihood-based pruning and consensus-based pruning, judged in terms of average number of links and nodes per correct word as a function of lattice accuracy.

As illustrated in Figure 1, the alignment is itself equivalent to a lattice, which we refer to as a *confusion network*. The confusion network has one node for each equivalence class of original lattice nodes (plus one initial/final node), and adjacent nodes are linked by one edge per word hypothesis (including the null word).

We can think of the confusion network as a highly compacted representation of the original lattice with the property that all word hypotheses are totally ordered.

If in each cluster we add the path probabilities for each word, and pick the hypothesis with highest total probability (or no word at all if deletion at that point has the highest posterior), and concatenate them, we get the *consensus* sentence hypothesis. We have shown in [4] that on the Switchboard task the *consensus* hypothesis results in significant improvements over the MAP baseline approach:

Hypothesis	Word Error Rate(%)
MAP	38.5
Consensus	37.1
$\Delta$ WER	-1.4

Typical word lattices contain links with very low posterior probability. An important observation at this point is that such links are negligible in computing the total posterior probabilities of word hypotheses. We found that the results for the consensus hypothesis obtained when only 10% of links in the original lattice were retained were almost the same as the results of the consensus hypothesis when no pruning was involved. Therefore we can discard all the links with a low posterior probability when compared with the sum of the posterior probabilities of the links found in a cross-section of the lattice (i.e. total posterior probability). The cluster initialization and subse-

quent merging only considers links that survive the initial pruning. As such, the confusion networks have other interesting uses besides word error minimization. Next we show how they can be used for lattice compression.

### 3 LATTICE PRUNING VIA INTERSECTION

The first step in the consensual post-processing method is to eliminate all the low likelihood links in the original lattices. We prune a link if its likelihood falls more than some empirically determined threshold below the total likelihood. This link pruning step leads to a new representation with fewer words, i.e. containing only high likelihood words. We can view this entire process as combining all the high likelihood words to get a new set of hypotheses. There are two ways to manipulate this new set which contains hypotheses from the original lattice as well as other hypotheses formed by concatenating likely word hypotheses. One method is to retain only paths that existed in the original representation. We used the AT&T FSM Toolkit [6] to intersect the confusion networks obtained for different pruning thresholds with the original lattices. We ran this experiment on the Switchboard LVCSR task. The Switchboard corpus contains spontaneous, casual speech telephone conversations. It is known that the word error rates increase as the speech becomes less constrained and the acoustic conditions deteriorate, thus Switchboard is among the tasks with the highest word error rates. We carried out experiments on the set of lattices corresponding to the set of 2427 utterances from 14 conversations that formed the dev-test at the Johns Hopkins University LVCSR Workshop which has a baseline WER of 38.5%. Figure 2 shows the decrease in the average number of links (*AvgL*) and average number of nodes (*AvgN*)

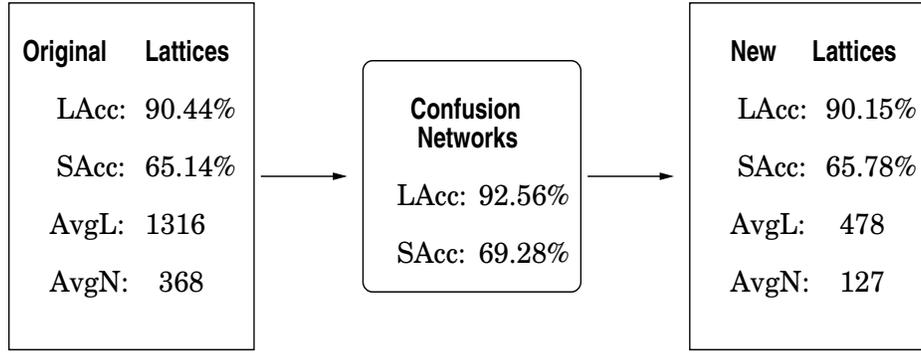


Figure 3: A new decoding pass on the confusion networks results in a set of smaller lattices (AvgN and AvgL are the average density nodes and links respectively) with similar accuracy (LAcc is lattice word accuracy and SAcc is lattice sentence accuracy).

per correct word versus lattice accuracy for both our consensus and likelihood-based pruning methods. The lattice accuracy is measured as the accuracy of the best path in the lattice. It can be seen that the consensus-based pruning is much more efficient than the likelihood-based method. For example, the following formulae show the differences in the average link density  $AvgL$  and average node density  $AvgN$  between the two pruning methods at a 87% lattice accuracy level:

$$Consensus\ AvgL = \frac{Likelihood\ AvgL}{2.12}$$

$$Consensus\ AvgN = \frac{Likelihood\ AvgN}{3.26}$$

And the reduction in number of links and nodes from the original representation:

$$Consensus\ AvgL = \frac{Original\ AvgL}{4.31}$$

$$Consensus\ AvgN = \frac{Original\ AvgN}{5.03}$$

The method is even more beneficial for some time and memory-consuming applications that are not overly sensitive to some decrease in accuracy. For example, if we consider 82%<sup>1</sup> a safe lattice accuracy level, the difference between the two pruning methods is even more dramatic:

$$Consensus\ AvgL = \frac{Likelihood\ AvgL}{4.16}$$

$$Consensus\ AvgN = \frac{Likelihood\ AvgN}{5.34}$$

And when compared with the original lattices:

$$Consensus\ AvgL = \frac{Original\ AvgL}{16.22}$$

<sup>1</sup>We ran experiments only down to 82% lattice accuracy

$$Consensus\ AvgN = \frac{Original\ AvgN}{17.14}$$

#### 4 LATTICE COMPRESSION VIA ACOUSTIC RESCORING

In the process of building the confusion network we connected words that were disconnected in the original lattice, i.e. we introduced new hypotheses. To assess the goodness of the newly introduced paths we compared the accuracy of the original lattices with the accuracy of the confusion networks, and found that we can obtain lattices with even better accuracy than the original ones, which suggests that some of the new sentence hypotheses are better than any of the previous ones. In the previous section we described a method for pruning lattices in which we discard all the new paths. Given that the confusion networks are more accurate than the initial lattices, we would like to keep all the new hypotheses and run another recognition pass constraining the search with the new representation. These new good paths disappeared in the previous decoding step due to either search errors or problems in the acoustic or language model. An ideal scenario would be to employ better but perhaps more expensive acoustic and language models in the new decoding process which would better discriminate between good and bad hypotheses, so as to take advantage of the new good hypotheses presented to it. This is something we intend to explore in the future. An easier experiment is to use exactly the same models on this new structure hoping that the new good hypotheses will get scores which will propagate them in the top candidates and the bad new hypotheses will be filtered out. We ran this experiment on the same WS97 dev-test set of lattices and found that the resulting lattices have almost the same word accuracy and slightly better sentence accuracy in the same time being three times smaller (see Figure 3). Sentence accuracy is the percentage of lattices which contained the correct sentence as a possible path.

## 5 CONCLUSION

This paper described a new way of utilizing the outcome of the consensual post-processing. In [4] we described an algorithm for transforming the original lattice into a different topology graph, the *consensus network*, and showed that using the new improved posterior probability estimates for word hypotheses we obtain significant improvements over the baseline MAP hypothesis on Switchboard. This paper shows an additional benefit that can be obtained from this method, namely the ability to efficiently discard unlikely paths based on the posterior probability of the word components. We report significant decrease in lattice size when compared with a likelihood-based pruning method. We also show how an additional decoding pass can reduce the size of the lattice by a factor of three with no loss in accuracy.

## 6 ACKNOWLEDGEMENTS

We thank Andreas Stolcke and Sanjeev Khudanpur for useful comments and suggestions. The work reported here was supported by NSF under NSF grant IRI-9618874 and IRI-9502312. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the funding agencies.

## References

- [1] L. R. Bahl, F. Jelinek, and R. L. Mercer. A maximum likelihood approach to continuous speech recognition. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume PAMI-5, pages 179–90, March 1983.
- [2] J. E. Hopcroft and J. D. Ullman. Introduction to Automata Theory, Languages, and Computation. In Addison-Wesley, Reading, MA, 1979.
- [3] L. Mangu, E. Brill, and A. Stolcke. Searching for Consensus to Improve Recognizer Output. In *Proc. Hub-5 Conversational Speech Recognition Workshop*, Linthicum, MD, September 24-25, 1998.
- [4] L. Mangu, E. Brill, and A. Stolcke. Finding Consensus among words: Lattice-based word error minimization. To appear in *Proc. Eurospeech-99*, Budapest, Hungary, 1999.
- [5] M. Mohri and M. Riley. Weighted determinization and minimization for large vocabulary speech recognition. In *Proc. Eurospeech*, vol. 1, pp. 131-134, Rhodes, Greece, 1997.
- [6] M. Mohri, F. Pereira, and M. Riley. FSM Library - general-purpose finite-state machine software tools. <http://www.research.att.com/sw/tools/fsm/>
- [7] H. Ney and X. Aubert. A word graph algorithm for large vocabulary, continuous speech recognition. In *Proc. ICSLP*, pp. 1355-1358, Yokohama, 1994.
- [8] J. Odell. The Use of Context in Large Vocabulary Speech Recognition. Ph.D. thesis, Cambridge University Engineering Department, Cambridge, U.K., 1995.
- [9] R. Schwartz and S. Austin. A comparison of several approximate algorithms for finding multiple (N-BEST) sentence hypotheses. In *Proc. ICASSP*, vol. 1, pp. 701-704, Toronto, 1991.
- [10] A. Stolcke, Y. Konig, and M. Weintraub. Explicit Word Error Minimization in N-best List Rescoring. In *Proc. Eurospeech-97*, pp.163-165, Rhodes, Greece, 1997.
- [11] F. Weng, A. Stolcke, and A. Sankar. Efficient Lattice Representation and Generation. In *Proc. ICSLP-98*, Sydney, Australia, 1998.
- [12] F. Weng, A. Stolcke, and A. Sankar. New developments in lattice-based search strategies in SRI's Hub4 System. In *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, pp. 138-143, Lansdowne, VA 1998.