

Confidence Based Lattice Segmentation and Minimum Bayes-Risk Decoding

Vaibhava Goel¹, Shankar Kumar², William Byrne²

¹I.B.M. T.J.Watson Research Center, Yorktown Heights, NY 10598

²Center for Language and Speech Processing, Johns Hopkins University, Baltimore, MD 21218

vgoel@us.ibm.com, skumar@clsp.jhu.edu, byrne@jhu.edu

Abstract

Minimum Bayes Risk (MBR) speech recognizers have been shown to yield improvements over the conventional maximum a-posteriori probability (MAP) decoders in the context of N-best list rescoring and A^* search over recognition lattices. Segmental MBR (SMBR) procedures have been developed to simplify implementation of MBR recognizers, by segmenting the N-best list or lattice, to reduce the size of the search space over which MBR recognition is carried out. In this paper we describe lattice cutting as a method to segment recognition word lattices into regions of low confidence and high confidence. We present two SMBR decoding procedures that can be applied on low confidence segment sets. Results obtained on the Switchboard conversational telephone speech corpus show modest but significant improvements relative to MAP decoders.

1. Introduction

A Minimum Bayes-risk (MBR) automatic speech recognizer attempts to find the hypothesis with the least expected error under a pre-specified task specific error measure. If $l(W, W')$ is the loss function that specifies the error between word strings W and W' , the MBR recognizer finds the optimal hypothesis as

$$\hat{W} = \operatorname{argmin}_{W' \in \mathcal{W}} \sum_{W \in \mathcal{W}} l(W, W') P(W|A), \quad (1)$$

where \mathcal{W} is the set of all word strings allowed in the language of the recognizer. In practice, \mathcal{W} has been taken to be the set of most likely recognition word strings, represented as an N-best list [4, 5] or lattice [9]. $P(W|A)$ is the posterior distribution on the word strings, usually obtained using an HMM acoustic model and an N-gram language model.

A simplification of the MBR recognizer can be derived under the assumptions that there is a unique segmentation of each string in \mathcal{W} into N substrings (of length zero or more) such that the total error between any pair W and W' can be decomposed into the errors over their corresponding substrings, i.e.

$$l(W, W') = \sum_{i=1}^N l_i(W_i, W'_i). \quad (2)$$

Let the segment set \mathcal{W}_i be the i^{th} set of substrings of the word sequences in \mathcal{W} . Equation 1 can now be re-written as follows.

$$\hat{W} = \operatorname{concat} \left[\operatorname{argmin}_{W'_i \in \mathcal{W}_i} \sum_{W_i \in \mathcal{W}_i} l(W_i, W'_i) P_i(W_i|A) \right]_{i=1}^N \quad (3)$$

which is simply a concatenation of the outputs of independent MBR decoders over each set \mathcal{W}_i . Here, it is also assumed that \mathcal{W} contains all the word strings that can be made by concatenating one string from each of \mathcal{W}_i . $P_i(W_i|A)$ is the marginal probability of W_i obtained by summing $P(W|A)$ for all those word

strings whose i^{th} substring is W_i . Furthermore, if $l_i(W_i, W'_i)$ is a 0-1 loss function, then Equation 3 further simplifies to

$$\hat{W} = \operatorname{concat} \left[\operatorname{argmax}_{W'_i \in \mathcal{W}_i} P_i(W_i|A) \right]_{i=1}^N \quad (4)$$

The simplification presented by Equation 4 has been utilized in several recently proposed N-best list and lattice based voting procedures to improve the recognition word error rate (WER) [6, 7, 8]. Specifically, these procedures produce a simultaneous word level alignment of all recognition hypotheses \mathcal{W} . This alignment trivially specifies a segmentation of \mathcal{W} into N sets \mathcal{W}_i . Since the alignment is produced at the word level, sets \mathcal{W}_i contain word strings of length zero or one. The loss function on each \mathcal{W}_i is the 0-1 loss function, which, in conjunction with the alignment, defines a sentence level loss function that approximates the word error rate. The voting hypothesis is selected according to Equation 4; it is the hypothesis with the least expected word error rate.

When the segment sets are restricted to contain single words, the alignments permitted between any two word sequences from \mathcal{W} may not include the optimal alignment associated with the Levenshtein string edit distance. As a result, voting procedures based on these restricted segment sets are not optimal with respect to the desired loss function, i.e the recognition WER. On the other hand, the more general procedures that implement Equation 1 directly suffer search errors in the case of large lattices.

Our goal is to explore intermediate solutions whereby the lattice is segmented, or cut, into sublattices containing word strings. The search on these lattice segments is performed under the exact cost between pairs of strings contained in these sublattices.

The rest of this paper is organized as follows. We first formalize our notion of lattice segmentation. We then derive lattice based quantities required for computing MBR hypotheses (Equation 3) over lattice segments. Finally we present our experimental results from lattice segmentation and compare them to results obtained by other MBR procedures.

2. A Formal Treatment of Lattice Cutting

A recognition lattice is a directed acyclic graph specified by the 5-tuple $(\mathcal{N}, \mathcal{E}, n_s, n_e, \rho)$; \mathcal{N} is the set of nodes, \mathcal{E} is the set of edges, n_s is the unique lattice start node, n_e is the unique lattice end node, and $\rho : \mathcal{N} \times \mathcal{E} \rightarrow \mathcal{N}$ specifies lattice connectivity. The directed acyclic nature of the lattice induces a partial order \leq on its nodes: $n_1 \leq n_2$ if there is at least one path connecting nodes n_1 and n_2 in the lattice and n_1 comes before n_2 on this path.

Let $(N_s \subset \mathcal{N}, N_e \subset \mathcal{N})$ be an ordered pair of lattice node sets such that

- P1. For all nodes $n \in \mathcal{N}$, there is at least one node $n' \in N_s$ such that $n \leq n'$ or $n' \leq n$.
- P2. For all nodes $n \in \mathcal{N}$, there is at least one node $n' \in N_e$ such that $n \leq n'$ or $n' \leq n$.
- P3. For any $n \in N_e$, there is no node $n' \in N_s$ such that $n \leq n'$.

Properties P1 and P2 essentially state that all lattice paths from lattice start to lattice end pass through at least one node of N_s and one node of N_e . Property P3 says that nodes of N_s on any lattice path precede nodes of N_e on that path. An example of N_s and N_e is depicted in the top panel of Figure 1.

Each lattice path can now be uniquely segmented into three parts by finding its first node that belongs to N_s and its first node that belongs to N_e . The portion of the path from n_s to the first node of N_s is the first segment; from the first node of N_s to the first node of N_e is the second segment; and from the first node of N_e to lattice end n_e is the third segment.

Segmentation of each lattice path, based on node sets $\{n_s\}, N_s, N_e, \{n_e\}$, defines a segmentation rule for dividing the entire lattice into three parts. In general, a rule for segmenting the lattice into $n + 1$ segments is defined by a sequence of lattice node sets $\{n_s\}, N_1, N_2, \dots, N_n, \{n_e\}$ such that all ordered pairs (N_i, N_{i+1}) , $i = 1, \dots, n - 1$ obey P1-P3. The i^{th} lattice segment, \mathcal{W}_{lat_i} , is specified by the node sets N_{i-1} and N_i . We shall say it is *bounded* on the left by N_{i-1} and on the right by N_i . An example lattice segment bounded by N_s and N_e is shown in the bottom panel of Figure 1. We note that while segmentation of lattice paths is based on some node sets, each segment retains the compact lattice format.

Lattice cutting yields sets \mathcal{W}_{lat_i} that are more constrained than those that could be obtained by explicitly enumerating all lattice paths and segmenting them. This is due to the sharing of nodes between lattice paths. For example, if we were to decide that the node labeled ‘‘O’’ belongs to N_s , then all lattice paths through that node must contain ‘‘WELL O’’ as their initial substring.

Despite its constraining nature, lattice cutting is attractive because it allows for efficient implementation of MBR search on each lattice segment. We now discuss this implementation in detail.

3. Segmental MBR Decoding of Lattice Cuts

On a lattice cut \mathcal{W}_{lat_i} , we wish to implement the MBR decoder of Equation 3, given as

$$\hat{W} = \underset{W'_i \in \mathcal{W}_{lat_i}}{\operatorname{argmin}} \sum_{W_i \in \mathcal{W}_{lat_i}} l(W_i, W'_i) P_i(W_i|A). \quad (5)$$

Here $P_i(W_i|A)$ is the marginal probability of W_i , obtained by summing over lattice paths whose segment belonging to \mathcal{W}_{lat_i} is W_i , i.e., the sum is over all those paths whose longest subpath in \mathcal{W}_{lat_i} is W_i . We note that $P_i(W_i|A)$ is in general different from the probability obtained by summing over all lattice paths that pass through W_i .

3.1. Marginal Probabilities of Paths in Lattice Cuts

We now introduce lattice cut related quantities that facilitate computation of marginal probabilities $P_i(W_i|A)$ from the acoustic and language model scores in the lattice, and consequently enable us to implement MBR search on lattice cuts.

Let W be a complete path in the lattice and let W_p be a prefix of W . We use $L_f(W_p)$ to denote the joint log-likelihood of observing W_p and the acoustic segment that corresponds to W_p . $L_f(W_p)$ can be obtained by summing the log acoustic and language model scores present on the lattice links that correspond to W_p . Similarly, for a suffix W_s of W , we use $L_b(W_s)$ to denote the joint log-likelihood of observing W_s together with its corresponding acoustic segment, conditioned on the starting node of W_s . $P(A)$ can be computed as $e^{L_f(n_e)}$.

Let $E^h(W')$ denote the first node of an arbitrary lattice path segment W' . Let $E^f(W')$ denote the last node of W' , and let $E(W')$ be the set of all lattice nodes through which W' passes, including $E^h(W')$ and $E^f(W')$. Let W_i be a path in a lattice segment \mathcal{W}_{lat_i} bounded by node sets N_s and N_e . Let $n_1 = E^h(W_i)$ and $n_2 = E^f(W_i)$. We first define a *lattice forward probability* of n_1 , $F_l(n_1)$, which is the sum of partial path probabilities of all partial lattice paths ending at n_1 . That is,

$$F_l(n_1) = \sum_{W_p: E^f(W_p)=n_1} e^{L_f(W_p)}. \quad (6)$$

However, paths that pass through any node of N_s before they reach n_1 would contribute a segment longer than W_i to this cut. We exclude their probability by defining a *restricted forward probability* of n_1 , restricted by the node set N_s , as

$$F_l(n_1; N_s) = \sum_{W_p : \begin{array}{l} E^f(W_p) = n_1 \\ E(W_p) \cap N_s = \{n_1\} \end{array}} e^{L_f(W_p)}. \quad (7)$$

We also define *lattice backward probability* of the final node of W_i , using the backward log-likelihood $L_b(W_s)$, as

$$B_l(n_2) = \sum_{W_s: E^h(W_s)=n_2} e^{L_b(W_s)}. \quad (8)$$

Using the restricted forward probability of n_1 and lattice backward probability of n_2 , the marginal probability of W_i can be computed as

$$P_i(W_i|A) = \frac{1}{P(A)} F_l(n_1; N_s) P(W_i, A(W_i)|n_1) B_l(n_2), \quad (9)$$

where $A(W_i)$ denotes the acoustic segment corresponding to W_i .

We note that if the node set N_s is such that no lattice path passes through two nodes of N_s , the restricted forward probability of n_1 , $F_l(n_1; N_s)$, will be identical to its lattice forward probability $F_l(n_1)$. In this case, the marginal probability of W_i will be obtained by summing over all lattice paths that pass through W_i . This is the well known lattice *forward-backward probability* of W_i .

3.2. Rescoring Lattice Segments

To obtain the MBR hypothesis (Equation 5) from a lattice cut \mathcal{W}_{lat_i} we first generate an auxiliary lattice corresponding to this cut. This is done by removing \mathcal{W}_{lat_i} from the lattice and adding dummy start and end nodes. The dummy start node is connected to all nodes of N_s with links having score $\ln F_l(n_s; N_s)$. Similarly, the dummy end node is connected to all nodes of N_e with

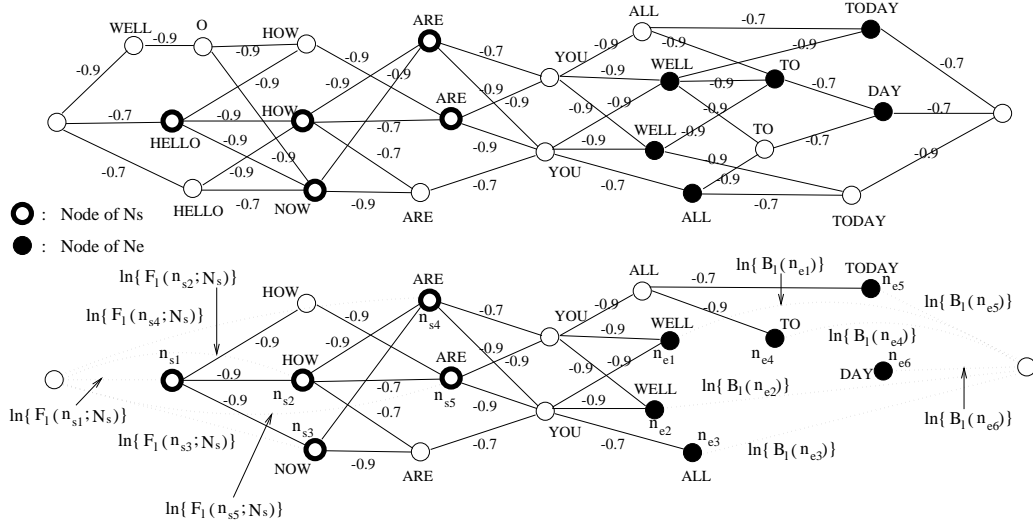


Figure 1: Cutting a lattice based on node sets N_s and N_e . The lattice segment bounded by these sets is shown in the bottom panel by solid line paths.

links having score $\ln B_l(n_e)$. An example auxiliary lattice is shown in the bottom panel of Figure 1.

An N-best list can be generated from each auxiliary lattice. Then, N-best rescoring [4, 5] or the extended ROVER (e-ROVER) procedure [2, 3] can be used to find the MBR hypothesis. We call the latter procedure lattice cutting e-ROVER (LCER). For admissible loss functions, an A^* search [9, 1] can also be performed on each auxiliary lattice.

4. Confidence Based Lattice Cutting for Levenshtein Loss Function

Lattice cutting based segmental MBR decoders can be implemented for any loss function for which the N-best rescoring or A^* search are feasible. The performance of these segmental decoders depends on how well the segmented loss (Equation 2) approximates the actual loss. In the following, we describe our preliminary attempts to find good lattice cuts for the Levenshtein loss function that measures recognition word error rate. Our procedure relies on finding parallel lattice links that are likely to align with each other when aligning any pair of lattice paths.

We start by identifying the one-best lattice path and compute the confidence score of each lattice link l on that path as follows.

1. Compute the lattice forward-backward probability of l .
2. Identify lattice links that have a time overlap of at least 50% with l . Among these links, keep only those that have the same word label as l .
3. Compute the lattice forward-backward probabilities of all these links. Add their probabilities to that of l to obtain the *confidence score* for l .

We identify high confidence links by comparing each link's confidence score to a global threshold. Consecutive high confidence links identify high confidence lattice regions and lattice cut node sets are derived as follows.

1. In any stretch of consecutive high confidence links, identify the left most link l_l and the right most link l_r .

2. Find all those lattice links that have a time overlap of more than 50% with l_l . The start nodes of these links form the left boundary node set of a lattice cut.
3. Find all those lattice links that have a time overlap of more than 50% with l_r . The end nodes of these links form the right boundary node set of a lattice cut.
4. Check for properties P1 and P2 and add nodes to the two boundary node sets to ensure that they are satisfied.

The top panel of Figure 2 depicts confidence based cutting of the lattice of Figure 1. The bottom panel shows the “determinized” version of the middle cut; the high confidence region can be represented by a single word sequence.

5. Experimental Results

All experiments reported here were performed on Switchboard LVCSR tasks.

The N-best list rescoring and lattice cutting e-ROVER procedures were tested on the Switchboard-2 portion of the 1998 Hub5 evaluation set (swbd2-98) and Switchboard-1 portion of the 2000 Hub5 evaluation set (swbd1-00). A description of the acoustic models and the language models used is given in CLSP LVCSR Hub5 system description [11]. The AT&T Large Vocabulary Decoder was used to generate an initial set of one-best hypotheses using HTK [12] cross-word triphone acoustic models, trained on VTN-warped data, with a 22K trigram language model. The one-best hypotheses were used to train MLLR transforms with two regression classes, for Speaker Adaptive Training (SAT) versions of the acoustic models. The decoder was then used to generate an initial set of lattices for the test set using MLLR and using pruned versions of SRI 33K trigram language model. The initial lattices were rescored using the unpruned SRI 33K trigram language model and then using SAT acoustic models with MLLR.

For lattice cutting, a confidence threshold of 0.9 was used. 250-best lists were generated from the auxiliary lattice for each cut. The likelihood scaling factor needed for rescoring was obtained in an unsupervised manner [5]; it was found to be 18.0. The results are reported in Table 1. These results were obtained

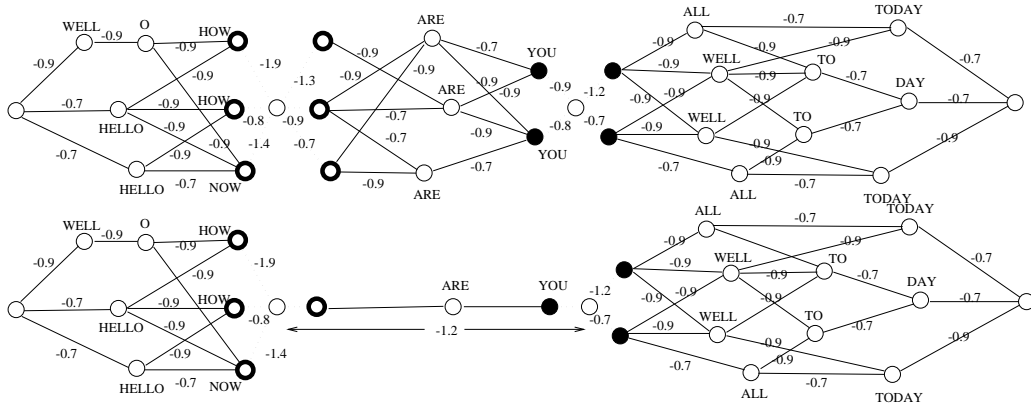


Figure 2: Pinching the lattice from Figure 1 into regions of low and high confidence based on word confidence scores.

in the development of the 2001 CLSP LVCSR Hub5 evaluation system.

Decoding Strategy		WER	
		swbd2-98	swbd1-00
MAP (baseline)		40.7	25.9
N-best rescoring	Entire lattice	40.1	25.5
	Lattice cutting	39.9	25.4
e-ROVER	Entire lattice	40.0	25.5
	Lattice cutting	39.7	25.2

Table 1: N-best rescoring and lattice cutting e-ROVER for 1998 and 2000 Hub5 evaluation sets.

The A^* search over lattice cuts was tested on the data set used at the 1997 Johns Hopkins University LVCSR workshop; full details of this set, including the lattice generation and the language models used, are given elsewhere [10]. A likelihood scale factor of 15.2 was used in the A^* search and a confidence threshold of 1.0 was used to cut the lattices. Results of the search are summarized in Table 2. The A^* procedure was not included in the 2001 CLSP Hub5 evaluation system, so results on this other test set are reported.

Decoding Strategy		WER
MAP (baseline)		38.5
N-best rescoring	Entire lattice	37.9
A^* search	Entire lattice	37.5
	Lattice cutting	37.1

Table 2: N-best rescoring and A^* search for 1997 JHU LVCSR workshop test set.

6. Conclusions

We have described a lattice cutting procedure that segments lattices into alternating regions of high confidence and low confidence. We have presented procedures for MBR decoding of these segmented lattices. Our experiments suggest that procedures such as N-best rescoring, e-ROVER and A^* search are improved when applied to these lattice cuts. Furthermore, we note that LCER is an improvement over lattice cutting N-best rescoring.

ACKNOWLEDGMENTS We thank Andreas Stolcke for the

use of the SRI language model and Michael Riley for the use of the AT&T Large Vocabulary Decoder.

7. References

- [1] Goel, V. and Byrne, W., "Minimum Bayes-Risk Automatic Speech Recognition", *Comp. Spch. & Lang.*, 14(2):115-135, 2000.
- [2] Goel, V., Kumar, S. and Byrne, W., "Segmental Minimum Bayes-Risk ASR Voting Strategies", *ICSLP-2000*, pp. 139-142.
- [3] Goel, V. "Minimum Bayes-Risk Automatic Speech Recognition", Ph.D. Dissertation, Johns Hopkins University, June 2001.
- [4] Stolcke, A., Konig Y. and Weintraub, M., "Explicit Word Error Minimization in N-Best List Rescoring", *Eurospeech 1997*, pp. 163-165.
- [5] Goel, V., Byrne, W. and Khudanpur, S. "LVCSR Rescoring With Modified Loss Functions: A Decision Theoretic Perspective", *ICASSP-1998*, pp. 425-428.
- [6] Fiscus, J. "A Post-processing System to Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduction (ROVER)", *IEEE Workshop on ASRU*, pp. 347-354.
- [7] Mangu, L., Brill, E. and Stolcke, A., "Finding Consensus Among Words: Lattice-Based Word Error Minimization", *Eurospeech-1999*, pp. 495-498.
- [8] Evermann, G. and Woodland, P., "Posterior Probability Decoding, Confidence Estimation and System Combination", *Proc. of the NIST Speech Transcription Workshop*, 2000.
- [9] Goel, V. and Byrne, W., "Task dependent loss functions in speech recognition: A^* search over recognition lattices", *Eurospeech-1999*, pp. 1243-1246.
- [10] Proceedings of the 1997 JHU LVCSR Workshop, <http://www.clsp.jhu.edu/ws97>.
- [11] The CLSP March 2001 Hub-5 Conversational Speech Transcription System, *Proc. of the NIST Speech Transcription Workshop*, 2001.
- [12] Young, S et.al, *The HTK Book*, Version 3.0, July 2000.