

Corrections to “Segmental Minimum Bayes-Risk Decoding for Automatic Speech Recognition”

Vaibhava Goel, Shankar Kumar, and William Byrne, *Member, IEEE*

Abstract—In our recently published paper [1], we presented a risk-based lattice cutting procedure to segment ASR word lattices into smaller sub-lattices as a means to improve the efficiency of Minimum Bayes-Risk (MBR) rescoring. In the experiments reported [1], some of the hypotheses in the original lattices were inadvertently discarded during segmentation, and this affected MBR performance adversely. This note gives the corrected results as well as experiments demonstrating that the segmentation process does not discard any paths from the original lattice.

I. INTRODUCTION

The purpose of this note is to correct and expand upon the experimental results presented in our recently published paper [1]. In Section III-B [1], we present a risk based lattice cutting procedure to segment ASR word lattices into sequences of smaller sub-lattices. The purpose of this procedure is to restructure the original lattice to improve the efficiency of Minimum Bayes-Risk (MBR) and other lattice rescoring procedures. Given that the segmented lattices are to be rescored, it is crucial that no paths from the original lattice be lost in the segmentation process. In the experiments reported in our original publication, some of the original paths were inadvertently discarded from the segmented lattices. This affected the performance of the MBR results presented. In this note we briefly review the segmentation algorithm and explain the flaw in our previous experiments. We find consistent minor improvements in Word Error Rate (WER) under the corrected procedure. More importantly, we report experiments confirming that the lattice segmentation procedure does indeed preserve all the paths in the original lattice.

II. RISK-BASED LATTICE CUTTING

Risk based lattice cutting is done by first aligning the lattice against the Maximum A Posteriori (MAP) hypothesis, and then using this alignment to identify candidate node sets for lattice segmentation. Periodic Lattice Cutting (PLC) selects a subset of these node sets in an attempt to balance MBR search errors over the lattice segments with the errors in approximating the loss function.

Figure 1 shows an example word lattice and the sub-lattices obtained by segmenting this lattice using PLC with a period of 1. The lower panel of this figure shows the N-best lists

generated from each lattice segment. As can be seen, lattice segments and N-best lists can contain deletions (shown as ϵ paths) relative to the MAP hypothesis. The role of ϵ paths is to allow the SMBR decoder to delete word sequences in the MAP hypothesis. In the risk-based lattice cutting experiments reported in the paper (Table 1 and Figure 14 in [1]), these ϵ paths were inadvertently removed from the lattice segments. This prevents the Segmental MBR decoder from picking these shorter lattice paths.

To assess how important it is to retain these ϵ paths, we first count the number of lattice segments that contain ϵ paths. Table I shows the percentage of lattice segments containing ϵ paths on the SWB2 held out set; this percentage is measured at various cutting periods used in PLC. We observe that ϵ -containing segments constitute a significant proportion of the lattice segments at period-1, but become negligible at periods 6 and above.

Cutting Period	1	2	4	6	≥ 8
	26.56	6.83	0.27	0.02	0

TABLE I

PERCENTAGE OF LATTICE SEGMENTS THAT CONTAIN ϵ PATHS AT VARIOUS PERIODS USED IN PERIODIC LATTICE CUTTING.

We next perform an experiment to verify that the risk based lattice segmentation does not discard any hypotheses in the original lattice. For each utterance we concatenate the sub-lattices produced by lattice segmentation to generate a new search space; we will refer to this new space as the *pinched lattice*.

We measure the Oracle-best Word Error Rate (OWER) of the pinched lattice as a function of the cutting period used in PLC; this is done both with and without allowing ϵ paths within lattice segments (Figure 2). OWER is defined as the word error rate of the hypothesis in the lattice that has the lowest Levenshtein distance relative to the reference transcription. On the SWB2 held out set, the OWER over the original lattices is found to be 24.1%. When ϵ paths are not allowed within lattice segments, the pinched lattices at period-1 obtain a higher OWER relative to the unsegmented lattices. At higher cutting periods, OWER of the pinched lattices is lower than that of the unsegmented lattices. In contrast, when ϵ paths are hypothesized within lattice segments, the OWER of the pinched lattices is lower than that of the unsegmented lattices at all cutting periods.

At lower cutting periods (1 and 2), there is a higher proportion of the lattice segments containing ϵ paths (Table I);

V. Goel is with the IBM T. J. Watson Research Center, Yorktown Heights, NY 10598 USA (e-mail: vgoel@us.ibm.com).

S. Kumar and W. Byrne are with the Center for Language and Speech Processing, Department of Electrical and Computer Engineering, The Johns Hopkins University, Baltimore, MD 21218 USA (e-mail: skumar@jhu.edu; byrne@jhu.edu).

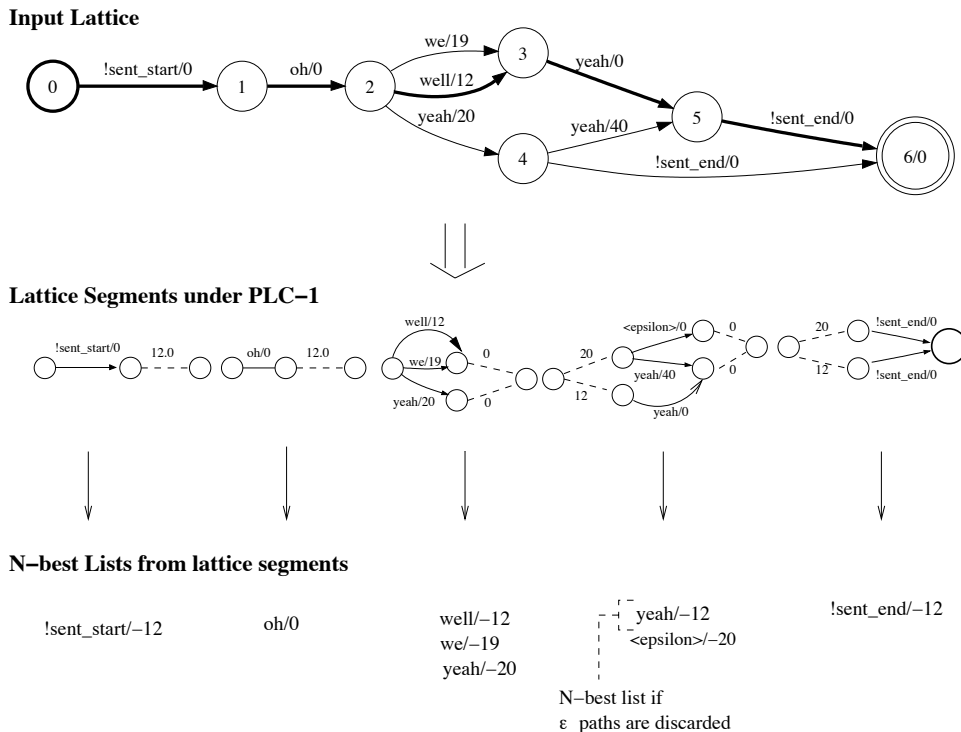


Fig. 1. Lattice segments obtained by applying Periodic Lattice Cutting (period=1) to a word lattice. The MAP hypothesis in the lattice is shown in bold. The lower panel shows the N-best lists generated from each lattice segment. Discarding ϵ paths from the lattice segments forces all paths in the pinched lattice to contain *yeah* as the final word.

therefore if we exclude these ϵ paths the pinched lattice contains fewer complete paths relative to the original lattice. This, in turn, leads to a degradation in the OWER of the pinched lattices relative to the original lattices. When ϵ paths are allowed with segments, the pinched lattice contains additional paths relative to the original lattice and therefore obtains a lower OWER. At higher cutting periods, there is only a small proportion of the segments that contain ϵ paths (Table I). In this case, excluding ϵ paths within segments does not affect the OWER of the pinched lattices. We note that when ϵ paths are allowed, the pinched lattices always yield a lower OWER than the original lattices; in contrast, other lattice-processing procedures such as confusion network generation [3] have been shown to degrade OWER of lattices [4].

III. PERFORMANCE OF SMBR RECOGNIZERS

We now present SMBR decoding experiments. We first determine the optimal cutting period in PLC. We try cutting periods of 1 through 14 and then generate the SMBR hypothesis using one of A^* search, N-best list rescoring, or e-ROVER procedures. Figure 3 shows the WER of A^* SMBR decoder on the SWB2 test set; this figure replaces Figure 14 [1]. We show the performance of PLC both with and without including ϵ paths within lattice segments. We observe that the optimal cutting period is 6 in both cases.

Table II presents a comparison of different lattice segmentation and hypothesis generation procedures. PLC is performed with a cutting period of 6. We show the performance of the RLC and PLC procedures both with and without including

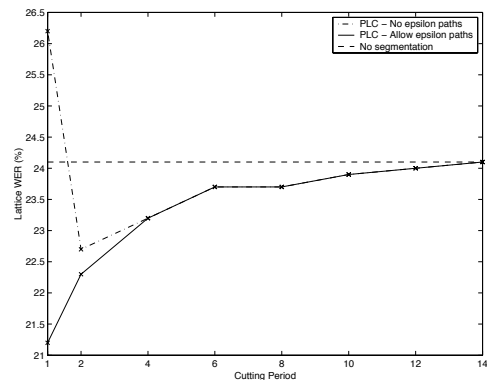


Fig. 2. Oracle-best Word Error Rate (OWER) of pinched lattices as a function of the cutting period used in Periodic Lattice Cutting. Results are shown on the SWB2 held out set. If ϵ paths are retained within lattice segments, the pinched lattices under PLC-1 and PLC-2 contain additional paths relative to the original lattices and their OWER is lower. If ϵ paths are discarded from the lattice segments, the pinched lattices under PLC-1 contain fewer paths relative to the original lattices and their OWER is higher.

ϵ paths in lattice segments. The values in bold represent the new results relative to Table 1 in the paper. The A^* MBR decoder [2], in its current implementation, is unable to perform MBR decoding over lattice segments containing ϵ paths. On these segments, we generate N-best lists (of size 250) and rescore the lists under the N-best MBR implementation [2].

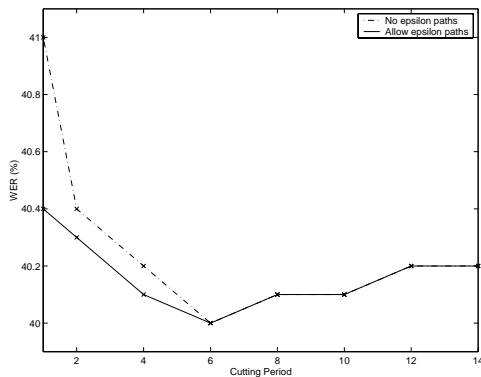


Fig. 3. Performance of periodic lattice cutting with different cutting periods for A^* SBR decoding on the SWB2 held out set.

Decoder		WER(%)	
		SWB2	SWB1
MAP (baseline)		41.1	26.0
SMBR Decoding			
Segmentation Strategy	MBR Decoding Strategy		
No Segmentation	A^* search	40.4	25.5
	e-ROVER	40.5	25.7
	N-best rescoring	40.4	25.6
ϵ paths not allowed in lattice segments			
RLC (Period 1)	A^* search	41.0	25.9
	e-ROVER	40.5	25.7
	N-best rescoring	41.0	25.9
PLC (Period 6)	A^* search	40.0	25.4
	e-ROVER	39.9	25.3
	N-best rescoring	40.1	25.4
ϵ paths allowed in lattice segments			
RLC (Period 1)	A^* search	40.4	25.5
	e-ROVER	40.2	25.5
	N-best rescoring	40.3	25.6
PLC (Period 6)	A^* search	40.0	25.4
	e-ROVER	39.9	25.3
	N-best rescoring	40.1	25.4
Confidence based	e-ROVER	40.0	25.2
	N-best rescoring	40.2	25.4

TABLE II

PERFORMANCE OF SINGLE SYSTEM SMBR SPEECH RECOGNIZERS.

IV. CONCLUSIONS

Our experiments show that when ϵ paths are allowed, the segmental MBR decoder under the RLC (PLC-1) segmentation achieves comparable performance relative to the MBR decoder on unsegmented lattices. This shows that PLC-1 segmentation does provide a good approximation of the Levenshtein loss function. Allowing ϵ paths leads to a substantial improvement in the performance of SMBR decoders under PLC-1. This shows that it is important to allow deletions during SMBR decoding. However, the performance of PLC at higher periods is unaffected by allowing ϵ paths. This is because at higher cutting periods, there are very few lattice segments that contain ϵ paths (Table I).

REFERENCES

- [1] V. Goel, S. Kumar, and W. Byrne, "Segmental Minimum Bayes-Risk Decoding for Automatic Speech Recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 12(3), pp. 287–333, 2004.

- [2] V. Goel and W. Byrne, "Minimum Bayes-risk automatic speech recognition," *Computer Speech and Language*, vol. 14(2), pp. 115–135, 2000.
- [3] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks," *Computer Speech and Language*, vol. 14(4), pp. 373–400, 2000.
- [4] G. Tur, J. Wright, A. Gorin, G. Riccardi, and D. Hakkani-Tur, "Improving spoken language understanding using confusion networks," in *ICSLP 2002*, Denver, CO, USA, 2002.