Published in ISCA ITRW Workshop on *Automatic Speech Recognition: Challenges for the new Millenium,* Paris, pages 47–52, sep 2000

Improved ROVER using Language Model Information

Holger Schwenk and Jean-Luc Gauvain

{schwenk,gauvain}@limsi.fr
LIMSI-CNRS, BP 133
91403 Orsay cedex, FRANCE

ABSTRACT

In the standard approach to speech recognition, the goal is to find the sentence hypothesis that maximizes the posterior probability of the word sequence given the acoustic observation. Usually speech recognizers are evaluated by measuring the word error so that there is a mismatch between the training and the evaluation criterion. Recently, algorithms for minimizing directly the word error and other task specific error criterions have been proposed. This paper presents an extension of the ROVER algorithm for combining outputs of multiple speech recognizers using both a word error criterion and a sentence error criterion. The algorithm has been evaluated on the 1998 and 1999 broadcast news evaluation test sets, as well as the SDR 1999 speech recognition 10 hour subset and consistently outperformed the standard ROVER algorithm. The approach seems to be of particular interest for improving the recognition performance by combining only two or three speech recognizers achieving relative performance improvements of up to 20% compared to the best single recognizer.

1. INTRODUCTION

During the last years the processing power of commonly available platforms has continuously increased leading to the use of very large acoustic and language models associated with sophisticated decoding algorithms in large vocabulary continuous speech recognizers. Alternative approaches have also become feasible, such as combining the outputs of several, possibly less performant but fast, continuous speech recognizers. The best such known approach was proposed by NIST in 1997 and named ROVER (*Recognizer output voting error reduction*) [2]. ROVER was first used to combine the results submitted by all participants in the LVCSR 1997 Hub 5-E evaluation: the word error rate was reduced from 44.9% (for the best single system) to 39.4%. This approach has since gained increasing interest with five of the nine participants in the 1998 broadcast news evaluation submitting a speech recognizer that itself is a combination of several different recognizers. Despite this, NIST was still able to reduce the word error rate from 13.5% to 10.6% by performing ROVER on outputs of the nine participating systems [7]. ROVER was also succesfully used in the 1999 broadcast news evaluation where a relative improvement of about 16% in the word error rate was observed [8].

Recently, links of the ROVER algorithm with theoretical work on n-best-list or lattice-based word error minimization [6, 9] and task-dependent error measures [5] have been established. To the best of our knowledge, however, there has been no implementation and large scale evaluation of a modified ROVER algorithm. We believe that there are many open questions, for instance, how important is the combination order of the system hypotheses ? how many systems should be combined ? is it advantageous to preprocess or normalize the systems' outputs prior to combination ? A new algorithm is presented that takes advantage of language model information during the decision process combining by these means a word and a sentence error criterion. This modification consistently improved performance on the broadcast news 1998 and 1999 evaluation set as well as on the SDR recognition task.

The next section summarizes the ROVER algorithm, and Section 3 presents some useful modifications when applying the algorithm in practice. Section 4 describes the algorithm which incorporates language model information, and the experimental results are summarized in Section 5.

2. ROVER

ROVER was developed by J. Fiscus of NIST [2]. It seeks to reduce word error rates for automatic speech recognition by exploiting differences in the nature of the errors made by multiple speech recognizers. ROVER proceeds in two stages: first the outputs of several speech recognizers are aligned and a single word transcription network (WTN) is built. The second stage consists of selecting the best scoring word (with the highest number of votes) at each node. The decision can also incorporate word confidence scores if these are available for all systems.

It is quite difficult to optimally align more than two word sequences and an iterative procedure is used. First, two sequences are aligned, creating a combined word transcription network. This WTN is aligned with the third word sequence giving a new combined word transcription network, that itself is aligned with the fourth word sequence and so on. The use of no-cost word transitions ("@"-arcs) allows insertions and deletions to be handled (see [2] for more details). Note that decisions are made **separately** at each node based on local information, i.e. the number of occurrences and/or the confidence score of each alternative arc. This means in particular that no information about the word context is used and as a result the combined output may have a very high perplexity. This is in contrast to the usual approach to speech recognition where language model (LM) information tends to reduce the perplexity of the hypotheses.

3. ANALYSIS AND EXTENSIONS

Table 1 gives the results of all the participants in the 1998 DARPA broadcast news evaluation [7]. The column labeled ROVER is the result of NIST combining all the nine systems in *alphabetical* order. Recall that four of the five best systems already used ROVER (ibm, cu-htk, dragon and bbn).

ibm	limsi	cu htk	dragon	bbn	philips rwth	sprach	sri	ogi fonix	nist **ROVER**
13.5	13.6	13.8	14.5	14.7	17.6	20.8	21.1	25.7	**10.6**

Table 1: Official word error rates in % for the 1998 broadcast news evaluation set (after [7]).

Order of combination

It is known that the pairwise alignment procedure of ROVER is to some extent affected by the order of combination. Furthermore, ROVER is here used to combine outputs of continuous speech recognizers, that means a sequence of words without any sentence structure. For efficiency reasons, during the alignment process it is necessary to split one document into smaller parts (for broadcast news, each document contains more than 14k words). This is done by searching for gaps larger than one second in the first word sequence. The document is then split at this point if there is a corresponding silence in all other word sequences. Obviously, the results depend on which word sequence is used first. Therefore, it can be advantageous to use the best single word recognizer as the first system, and more generally, to combine them in the order of decreasing recognition rate.

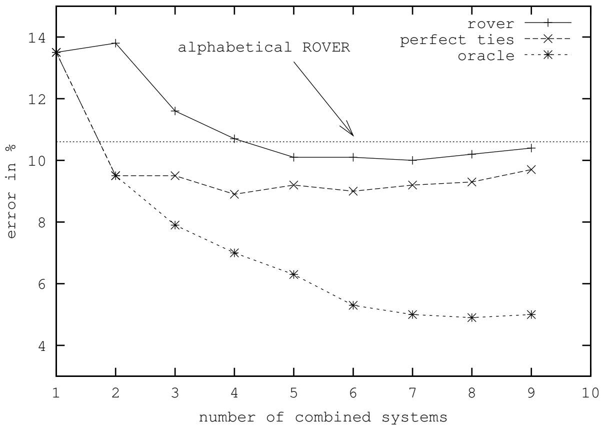

Figure 1: 1998 broadcast news word error rates in function of the number of combined systems (individual error ranked order): deciding ties arbitrarily („rover”); making the best choice among the ties („perfect ties”) and using the best fitting sequence in the whole aligned WTN („oracle”).

Figure 1 (solid line) shows the word error rates when the recognizers are combined in error ranked order. Although the combination of nine systems in ranked order instead of alphabetical order achieves only a very slight reduction in word error to 10.4%, a minimum word error of about 10.1% can be obtained when combining 5 to 8 systems. It appears that combining many systems, in particular those with higher error rates, is of no benefit and may actually increase the error rate of the combined system.

Normalization/filtering

The standard NIST scoring procedure applies a filtering/ normalization of the recognizer's output prior to alignment with the reference transcription. This normalization includes mappings of alternative spellings to one common form (e.g. `afterall` → `after all`, `cannot` → `can not`, ...), and mappings of abbreviated forms to several variants (e.g. `CHILD'S` → `CHILD'S` or `CHILD IS` or `CHILD HAS`). We suggest applying this filtering **before** combining the systems with ROVER. The alignement of word sequences with variants, however, is not easy to incorporate into the ROVER algorithm so that only the one-to-one filtering rules were applied. There is only a slight decrease in the word error rate (10.1% to 10.0%) when combining the outputs of 7 recognizers. The application of all the filtering/normalization rules may lead to larger performance improvements.

Treatment of ties

When combining the outputs of several systems it is quite frequent that after alignment some words appear equally often at a given node in the WTN and an arbitrary decision has to be taken (see Table 2). These ties could be broken using confidence scores for the transcriptions of the individual systems, but unfortunately only three of nine participants of the 1998 broadcast news evaluation provided them, so that this option wasn't possible. Also, the confidences scores provided by different recognizers may be difficult to compare.

# of combined recognizers	2	3	4	5	6	7	8	9
# of ties	4726	1560	1539	987	1075	884	923	846

Table 2: Number of ties for 1998 broadcast news.

Instead, we determined the error rate that could be obtained if all ties were able to be correctly resolved (see Figure 1 upper dashed line). In this case the word error rate would be about 9% which would be a significant improvement. In the next section, an approach which using LM information is used to break ties. The upper bound on the performance that could be obtained with such an approach, i.e., the error rate that is achieved if the correct word at each branch were chosen among all the alternatives (oracle-mode), is shown in Figure 1 (lower dashed line). It can been seen that the combined transcriptions from the nine speech recognizers contains the correct word over 95% of the time. These results are of course only of hypothetical value, but it seems nonetheless that there is some hope for further improvement of the combination approach.

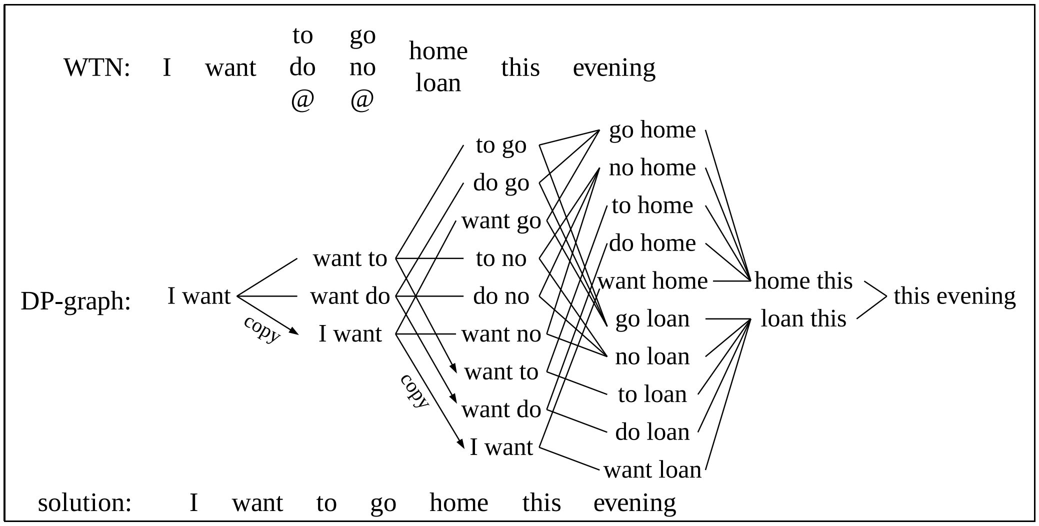

Figure 2: Exemple of execution of the modified dynamic programming (DP) algorithm for finding the word sequence with minimal perplexity when null-arcs are present. WTN is the word transition graph and the DP-graph shows a representation of all the nodes that are generated thru dynamic programming.

4. INCORPORATION OF LANGUAGE MODEL INFORMATION

One of the intrigues about the success of ROVER is that it seems to work well even though no contextual or language model information is used in the voting. In fact, it could theoretically happen that the resulting word sequence has a higher perplexity than any of the individual word sequences. Therefore, we propose using LM information to provide contextual information. This is done in the following way: first the outputs of all recognizers are aligned and the most likely word is selected at each branch of the word transition network. If several words are equally frequent, all of them are kept. Second the language model of the LIMSI broadcast news system is used to select the word sequence among all alternatives that minimizes the perplexity.

To the best of our knowledge, similar modifications of the reference ROVER algorithm have not been reported in the literature. There is however, related work on hypotheses selection during decoding for a single speech recognizer [5, 6, 9]. In the standard approach to speech recognition, the goal is to find the sentence hypothesis that maximizes the posterior probability $P(W|A)$ of the word sequence W given the acoustic observation A. Usually speech recognizers are evaluated by measuring the word error so that there is a mismatch between the training and the evaluation criterion. Recently, algorithms for minimizing directly the word error have been proposed [5, 6, 9]. These approaches have been evaluated on the Switchboard corpus and achieved a small but consistent decrease in word error and an *increase* of the sentence error, in accordance with the new optimization criterion.[1] It is believed that word error minimization is most effective on tasks with relatively high error rates since a wrong sentence probably contains several wrong words.

In contrast to the above cited approaches to hypothesis selection in a single speech recognizer output, only limited information is available when applying ROVER: one single transcription with timing information for each speech recognizer. The only information that can be used is the number of occurrences of each word at a given node in the WTN, which was demonstrated to lead to suboptimal results when ties are arbitrarily broken. Our proposal to use a LM to break these ties combines a word error oriented criterion (local number of occurrences) with a sentence error criterion (minimum perplexity of the global word sequence).

We have reimplemented the ROVER algorithm in order to incorporate language model information. The program can combine the outputs of the nine 1998 broadcast news systems in 0.01xRT on a SGI UNIX workstation. Development of alignment procedures that support variants or n-best lists as input is currently underway.

Dynamic Programming Algorithm

Figure 2 top shows an example of a WTN obtained after aligning several transcriptions. All nodes that are not ties have already been simplified by selecting the most frequent word. The remaining words will be decided by minimizing the perplexity of the overall sentence. Note that all possible sentences do not have the same length when the WTN contains null-arcs. In order to reduce the tendency to prefer short sentences, a fixed penality was applied each time a null-arc was used during the optimization process. The value of this penality was detemined on an independent development set. It has also turned out that its exact value is not very critical and the results on the three test corpora did not vary significantly for a wide range of the penalization factor.

Unfortunately, the standard dynamic programming solution to 3-gram LM perplexity optimization can not be used since the presence of multiple null-arcs prevents local 3-gram LM evaluation. In the example above, for instance, the 3-grams “`I want to`” and “`I want home`” have to

[1] Mangu et al. do not report sentence errors [6].

number of combined systems:	2	3	4	5	6	7	8	9
arbitrary ties:								
word error:	13.8%	11.6%	10.7%	10.1%	10.1%	10.0%	10.2%	10.4%
sentence error:	81.0%	76.3%	74.3%	73.0%	73.8%	73.4%	73.4%	74.6%
perplexity:	183.8	171.6	166.1	164.2	161.7	160.2	159.3	159.6
using LM to break ties:								
word error:	12.5%	11.1%	10.3%	10.1%	10.1%	10.0%	10.3%	10.5%
sentence error:	79.9%	75.4%	73.3%	72.6%	73.0%	72.9%	74.2%	74.7%
perplexity:	137.2	145.8	146.5	151.2	149.6	150.8	150.0	151.1

Table 3: 1998 broadcast news test set word error rates and perplexity when using LM information instead of braking ties arbitrarily NIST's ROVER achieves 10.6% word error and 73.7% sentence error.

be evaluated. Therefore the following extension of the standard dynamic programming (DP) algorithm was used:

1) build the DP start node using the first two words in the WTN (in this example: `I want`).

2) repeat until end of the WTN:

   - build DP-nodes as a combination of all words in the current WTN-node (for instance `to` and `do`) and all right words in the previous DP-node word pairs (e.g. `want` of the pair `I want`);
   - evaluate the 3-gram LM models between all corresponding DP-nodes, retaining only the minimum in the case of multiple entering arcs (e.g. all the arcs entering into the DP-node `go home`);
   - if the current WTN-node contains a null-arc, copy all previous DP-nodes and add the null-arc penality instead of evaluating a LM; (e.g. the null-arc in the WTN-node `go,no,@` results in copying all the previous DP-nodes).

3) backtrack to find the solution.

This dynamic programming algorithm achieves the usual complexity reduction. In the case of many WTN-nodes with null-arcs that follow each other, additional processing time is needed to copy and process the resulting DP-nodes. In all our experiments, the overall processing time of the DP-algorithm was several orders of magnitude lower than the one of a direct solution through extensive search. If the word sequence is longer, e.g. more than 10 words, the direct solution was in fact not feasible any more, but the proposed DP-algorithm runs in a fraction of real time. In the example of Figure 2 only 40 trigrams need to be evaluated in comparison to 108 when all possible sentences (including begin and end of sentence) are explored.

5. EXPERIMENTAL RESULTS

The modified ROVER algorithm and the benefit of incorporating language model information into the decision process have been evaluated on three large scale continuous speech recognition corpora. The 1998 and 1999 broadcast news evaluation test sets each contain approximatively three hours of speech of varying difficulty (studio quality, telephone, foreign speakers, ...) [7, 8]. Recognition results are also reported on a representive 10 hour subset selected by NIST from the TDT-2 audio corpus [1], and used in the 1999 and 2000 SDR evaluations [3]. The results are summarized in the following sections.

Results on Broadcast News 1998

Table 3 gives the improvements of the word and sentence error rate as well as the perplexity when using a LM to break ties. An interesting result is obtained when combining just two systems: 8.1% relative word error reduction with respect to the best two individual systems (13.5 and 13.6% word error rate respectively). Note that with only two systems ties always occur when the two systems disagree, which means that the LM is used for the whole decision process. Standard ROVER, e.g. breaking ties arbitrarily, does not work when combining just two systems (the word error increases to 13.8%). As can be seen in Figure 3, the use of LM information to break ties always gives a better result than taking an arbitrary decision, but it is of particular interest when only few recognizers are combined: for instance a word error rate of 11.1% is achieved when combining the three best recognizers.

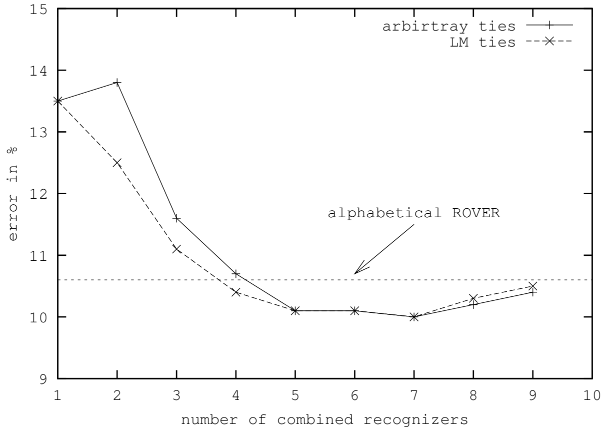

Figure 3: 1998 broadcast news word error rates when using LM information instead of breaking ties arbitrarily (see text for more details).

We did not observe any increase in the sentence error when using the original ROVER algorithm nor when incorporating LM information (see Table 3).

Results on Broadcast News 1999

The modifications of the ROVER algorithm were verified on the 1999 broadcast news evaluation test set. The focus of this evaluation was on 10xRT large vocabulary continuous speech recognizers. Table 4 summarizes the official results of the individual recognizers and of the reference ROVER run by NIST[2] [8].

	10x RT					50x RT
	LIMSI	BBN	IBM	NIST BBN	CMU	**ROVER**
Werr	17.1%	17.3%	17.6%	24.6%	26.3%	**14.4%**
Serr	77.2%	79.7%	78.4%	83.2%	83.8%	**73.8%**

Table 4: Official word and sentence error rates on the 1999 broadcast news evaluation test set.

The original ROVER achieves a relative word error reduction of 16% with respect to the best single 10xRT recognizer when used to combine the five 10xRT recognizers in alphabetical order. The ROVER also outperformed the two unlimited computation systems for which results were reported [8]. The LIMSI unconstrained system ran in 54xRT and obtained a word error of 15.9% and a sentence error of 75.6%. IBM's unconstrained system ran in 2000xRT and obtained a word error of 15.0% and a sentence error of 75.6%. This may indicate a new direction for future research in speech recognition: developing several fast recognizers and combining them may lead to better performance than one very complicated one.

number of combined systems:	2	3	4	5
arbitrary ties:				
word error:	18.9%	14.3%	14.1%	14.1%
sentence error:	80.9%	74.1%	73.4%	72.9%
arbitrary ties + LM:				
word error:	15.2%	**13.6%**	13.8%	14.0%
rel. improvement:	-11.1%	**-20.5%**	-19.3%	-18.1%
sentence error:	75.8%	73.4%	72.5%	73.0%
rel. improvement:	-1.8%	-4.9%	-6.1%	-5.4%

Table 5: 1999 broadcast news test set word and sentence error rates when using LM information compared to breaking ties arbitrarily. The relative improvement is indicated with respect to the best single recognizer (17.1% werr, 77.2% serr).

The large range in the word error rates (see Table 4) suggests combining only the three best recognizers. When these three recognizers are combined and a LM is used to break ties we achieve a word error of 13.6% in 30xRT. This is a 5.6% relative improvement with respect to the alphabetical ROVER (14.4% werr, 50xRT) and about 20% relative improvement with respect to the best individual recognizer (17.1% werr, 10xRT). This confirms our earlier observation that the modified ROVER works well when combining a small number of system outputs. Table 5 summarizes the results combining two to five 10xRT recognizers.

The order of combination should be determined using the performance of each 1999 recognizer on the previous year's test set (broadcast news 1998), but this information was not available for all recognizers at the time of writing this paper, so the actual word errors on the 1999 test set were used. However, only minor differences in the results with respect to the ordering of the recognizers are observed. We combined the three best recognizers in all possible orders: the average word error rate was 13.67% and the maximum word error rate was 13.74% (inverse order of the three best recognizers).

For comparison, the LM has been used on the whole WTN, that means disregarding all information on the number of occurrences of each word. As expected, the results were worse than when breaking ties arbitrarily (18.5% and 21.4% word error when combining the outputs of three or four recognizers respectively).

Results on SDR 1999

The transcription accuracy on the representative 10h test subset of the 1999 SDR data [3] is given in Table 6 for three speech recognizers: `cuhtk-s1su` and `nist-b1su` are speech recognizers used for the 1999 TREC evaluation [10]; `limsi-s2su` uses the same acoustic and language models as the 1999 system, but a new decoder [4].

	cuhtk-s1su	limsi-s2su	nist-b1su
original filtering:			
word error:	20.5%	21.3%	26.7%
sentence error:	95.3%	94.4%	95.0%
extended filtering:			
word error:	20.4%	20.0%	26.7%
sentence error:	95.2%	93.7%	94.9%

Table 6: Word and sentence error rates for the 1999 SDR 10h subset.

In contrast to broadcast news scoring, the filtering/normalization of NIST's standard SDR scoring procedure does not include rules for contractions like `I'M` → `I AM`. As can be seen in Table 6, these rules are very important for scoring LIMSI's recognizer while there is no significant difference for the other recognizers. We suppose that these recognizers already output the long form of many contractions. The results of the combined systems are summarized in Table 7 (order `cuhtk`, `limsi` and `nist`). All normalizing/filtering rules that do not generate several variants have been used.

The conclusion for the broadcast news evaluations test sets also holds for this larger test set. For instance, a very competitive system is obtained by combining the outputs of the first two speech recognizers: a 7.0% relative improvement in the word error with respect to the best single recog-

[2] This year, NIST also used normalizing/filtering prior to combination.

number of combined systems:	2	3
arbitrary ties:		
word error:	21.6%	18.0%
sentence error:	95.0%	92.4%
arbitrary ties + LM:		
word error:	18.6%	17.4%
rel. improvement:	-7.0%	-13.0%
sentence error:	93.3%	91.6%

Table 7: SDR 10h subset word and sentence error rates when using LM information compared to breaking ties arbitrarily. The relative improvement is indicated with respect to the best single recognizer.

nizer, while standard ROVER, i.e., breaking ties arbitrarily, does not work in this case. Combining three outputs results in even lower error rates, but the benefit of using language model information gets smaller. This can probably be explained by the fact that ties are less frequent when more recognizer outputs are combined.

6. CONCLUSION

This paper gives a detailed analysis of the behavior of the ROVER voting scheme on the 1998 and 1999 broadcast news evaluation set as well as on the SDR recognition task. Our experiments indicate that it may hurt performance if too many systems are combined, and that it is better to eliminate those with the highest error rates. Additional improvement can be obtained by filtering/normalizing the outputs of the different speech recognizers prior to combination by ROVER.

We have presented an extension of the ROVER algorithm that uses a language model to decide ties in the number of occurences of words in the word transition network. By these means a word error oriented criterion (local number of occurrences) is combined with a sentence error criterion (minimum perplexity of the global word sequence). This approach seems to be of particular interest for improving the recognition performance by combining only two or three speech recognizers: relative improvements of up to 20% with respect to the best single recognizer were obtained on several complicated broadcast news recognition tasks. In our experiments, the presented algorithm consistently outperformed the original ROVER algorithm.

REFERENCES

[1] C. Cieri, D. Graff, M. Liberman. The TDT-2 Text and Speech Corpus. In *Proc. DARPA Broadcast News Workshop*, pages 57–60, 1999. (See also http://morph.ldc.upenn.edu/TDT).

[2] J. G. Fiscus. A post-processing system to yield reduced error word rates: Recognizer output voting error reduction (ROVER). In *IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 347–354, 1997.

[3] J. S. Garofolo et al., 1999 Trec-8 Spoken Document Retrieval Track Overview and Results. In *Proc. 8th Text Retrieval Conference TREC-8*, Nov. 1999.

[4] J. L. Gauvain and L. Lamel. Fast decoding for indexation of broadcast data. In *ICSLP*, 2000.

[5] V. Goel and W. J. Byrne. Minimum Bayes-risk automatic speech recognition. *Computer Speech And Language*, 14(2):115–135, 2000.

[6] L. Mangu, E. Brill, and A. Stolcke. Finding consensus among words: Lattice-based word error minimization. In *Eurospeech*, pages 495–498, 1999.

[7] D. S. Pallett, J. G. Fiscus, J. S. Garofolo, A. Martin, and M. Przybocki. 1998 broadcast news benchmark test results: English and non-English word error rate performance measures. In *DARPA Broadcast News Workshop, Hernon, VA*, Feb. 1999.

[8] D. S. Pallett, J. G. Fiscus, and J. S. Garofolo. 1999 broadcast news benchmark test results. In *DARPA Broadcast News Workshop, Washington*, May 2000.

[9] A. Stolcke, Y. König, and M. Weintraub. Explicit word error minimization in n-best list rescoring. In *Eurospeech*, pages 163–165, 1997.

[10] In *Proceedings of the 8th Text Retrieval Conference TREC-8*, Nov. 1999.