ROBUST CONFIDENCE ANNOTATION AND REJECTION FOR CONTINUOUS SPEECH RECOGNITION

Benoît Maison and Ramesh Gopinath

IBM Thomas J. Watson Research Center
P.O. Box 218 – Yorktown Heights, NY 10598 – USA
{bmaison,rameshg}@us.ibm.com

ABSTRACT

We are looking for confidence scoring techniques that perform well on a broad variety of tasks. Our main focus is on word-level error rejection, but most results apply to other scenarios as well. A variation of the Normalized Cross Entropy that is adapted to that purpose is introduced. It is successfully used to automatically select features and optimize the word-level confidence measure on several test sets. Sentence-level confidence geared toward the rejection of out-of-grammar utterances is also investigated. The combination of a word graph based technique and the acoustic score shows excellent performance across all the tasks we considered.

1. INTRODUCTION

Reliable confidence measures are critical to make speech applications more usable when recognition accuracy is less than perfect. Even when accuracy is high, out-of-vocabulary words or non-speech events need to be detected.

We experiment with methods for estimating word-level confidence on a wide array of continuous speech recognition tasks, ranging from the large vocabulary broadcast news to the grammar-based car navigation commands. The focus is on acceptance/rejection scenarios, where the confidence measure is only used to ranks recognized words from least likely to most likely correct. We also look for methods that work well across all tasks with little or no training.

How to assess the quality of confidence measure is first discussed in section 2, and a variation of the widely used Normalized Cross Entropy (NCE) is proposed. The most effective tool in our arsenal is a lattice-based method similar to those proposed before [1, 2]. It is studied in more detail in section 3. The combination of lattice-based confidence with other techniques in order to achieve increased performance and robustness is examined in section 4. The same tools are finally applied to the rejection of out-of-grammar utterances in section 5.

The various test sets we are using are described in Table 1. The first two are publicly available, the others are internal.

2. ASSESSING CONFIDENCE MEASURES

Confidence annotation consists in associating with each word or phrase output by a speech recognizer a score that reflects the expected accuracy of the recognition. For some applications, like speech understanding systems [3], the confidence measure may be used as the probability of a word being correct, and should be evaluated as such. We call this case the "confidence as probability scenario".

However, in many – if not most – cases, a confidence measure is only used to determine whether a recognized word or sequence of words should be trusted or rejected. *Absolute* values do not matter, only ranks do, since words are accepted or rejected according to a threshold. The quality of the confidence measure resides in its ability to distinguish between correct and wrong words. We call this the "acceptance/rejection scenario".

The performance a confidence measure used for rejection is represented by its Receiver Operating Characteristic (ROC). Let N be the number of decoded words, q_i the estimated confidence of word i, and c_i its true status (0 means error, 1 means correct). The insertion+substitution rate is $p_e = \sum_{i=1}^{N} c_i/N$. For a given rejection threshold τ, the false acceptance rate is defined as $FA(\tau) = \sum_{i:q_i \geq \tau} (1 - c_i)/N$ and the false rejection rate is defined as $FR(\tau) = \sum_{i:q_i < \tau} c_i/N$. The curve $(FA(\tau), FR(\tau))_{\tau=-\infty}^{+\infty}$ summarizes the tradeoffs associated with the confidence measure q.

Note that since $FA(\tau)$ and $FR(\tau)$ are relative to the total number of words, not normalized against the number of errors or the number of correct words. Hence the curve intersects the axes for $FA = p_e$ and $FR = 1 - p_e$. Indeed, the value of a given acceptance/rejection tradeoff for a particular application depends on the balance between three types of errors: recognizer errors that are correctly detected, recognizer errors that are missed, and correct words that are wrongly rejected. Normalized acceptance/rejection rates would hide that information.

It is often useful to have a single measure of the performance of a confidence annotation technique. The Equal Error Rate is one such measure: it is the operating point where the (normalized or unnormalized) False Acceptance rate is equal to the False Rejection rate. It is the point of the ROC closest to the origin of the axes. Unfortunately, it is also the least interesting area of the curve for applications that need to detect the most likely correct words or the most likely errors. Those correspond to the tails of the ROC. Ideally, the operating point suited to the application should be used, but how to choose it is not always known ahead of time. Besides, confidence annotation techniques that can be used across many ap-

The authors would like to thank Lidia Mangu and Salim Roukos for their insightful comments and suggestions.

	Name	Channel	LM	Voc. size	WER	S+IR	#words
A	Broadcast News ’98	mixed	4-gram	68K	16.65	13.44	31166
B	Switchboard ’98	telephone	3-gram	29K	38.50	29.61	18393
B	Travel reservations	telephone	3-gram	3.3K	19.13	14.56	2754
D	Car environment	far field mic	FSG	11/127	2.53	2.01	50327
E	Stocks Names	cell phone	FSG	8.5K	24.4	19.02	4116
F	Car navigation	cell phone	FSG	20–20K	10.39	9.07	3141

Table 1. The test sets. From left to right: Label, Task name, Channel, Language modeling (either n-gram or finite-state grammar), Word Error Rate, Substitution+Insertion Rate, Number of Annotated Words.

plications are desirable.

The Normalized Cross Entropy introduced by NIST provides an interesting alternative. It is defined as the relative decrease in uncertainty brought by the confidence measure about the status of the words (correct or wrong) :

$$\text{NCE} \quad = \quad \frac{H_e + \frac{1}{N}\sum_{i=1}^{N}\left[c_i \log(p_i) + (1-c_i)\log(1-p_i)\right]}{H_e}$$

where $H_e = -p_e \log(p_e) - (1-p_e)\log(1-p_e)$. The confidence measure q_i has been replaced by p_i in the above formula to stress that NCE requires the confidence values to be also probability estimates. This may require an additional – and potentially difficult – step in the estimation procedure. In [2], a decision tree is used to map q_i into p_i. However, as long as this last step does not affect the ranking of the scores (i.e. the mapping is a monotone increasing function) it is irrelevant to all rejection scenarios. On the other hand, an attractive feature of NCE is that it gives a larger weight to words that are annotated with confidence close to zero or one.

In order to combine the advantages of both NCE and ROC, we propose to incorporate the optimal monotone increasing mapping $p_i = f^*(q_i)$ into the figure of merit. We define NMCE as

$$\text{NMCE}\{c_i, q_i\} = \sup_{f(\cdot)} \text{NCE}\{c_i, f(q_i)\}.$$

NMCE only depends on how the words are sorted from lowest to highest confidence, yet provides a single measure of performance. The function $f(\cdot)$ needs only to be defined for the values q_i. The following simple algorithm due to Ayer et al. [4] can be used for that purpose. First, all words that share the same confidence value q_i are grouped together. $f^0(q_i)$ is initialized to ratio of correct words in that set. Then all pairs of distinct values (q_i, q_j) that violate the monotonicity constraint $(f^0(q_i) > f^0(q_j), q_i < q_j)$ merge and $f^1(q_i) = f^1(q_j)$ are re-estimated as the ratio of correct words in the merged set. The procedure terminates in a finite number of steps and produces $f^*(\cdot)$. NMCE is NCE computed with $p_i = f^*(q_i)$ as defined above. We will use this figure of merit in the following sections to select features and optimize combinations of features.

In a scenario where the confidence has to be *used* as a probability estimate, a similar algorithm can be used to learn the mapping function from held out data instead of the test data. The function has to be further adjusted such that its output is inside the $]0,1[$ interval with some margin [2].

It should be noted that although NCE or NMCE provide some degree of normalization, they do not allow a fair comparison of confidence annotation techniques across recognition systems. The influence of the error rate was noted by [5], as well as the fact that word accuracy can be traded for NCE. One partial solution could be to compare confidence measures that are applied not only to the same speech data but also to the same fixed decoded script.

3. LATTICE-BASED METHODS

Methods based on word graphs have been shown to be the single most useful confidence measure available [1, 6], slightly outperforming similar-minded methods based on N-best lists. They proceed as follows. The set of hypotheses considered by the recognizer is represented as a directed graph with exactly one start node and one end node. Each arc $j \in [1 \ldots L]$ carries a word along with its score (denoted s_j, usually a mixture of acoustic and language model log-likelihoods: $s_j = \lambda a_j + l_j$), and a start and end time. Each possible path in the graph defines one hypothesis. The arc scores are first scaled and the exponential is taken: $e_j = \exp(\alpha s_j)$. A forward-backward procedure is then applied to come up with arc posteriors w_j. The forward-backward performs a normalization over all *complete path* probabilities. Provided α is strictly positive, the complete path probabilities still reflect the ordering determined by the recognizer. The posterior of an arc is the sum of the probabilities of all the paths that go through that arc. Next, frame-based confidence is computed. For each time frame, it is the sum of the posteriors of all arcs accounting for that frame that carry the same word as the recognized word accounting for that frame. Finally, the frame confidence over all frames spanned by each decoded word is combined to give a word-level confidence.

Lattice-based methods are attractive because they combine several features traditionally used for confidence annotation (acoustic and language modeling scores, number of competing words in N-best list, etc.) in a way that is consistent with how the recognizer used them in the first place. Another example of the power of word posteriors is that they can also be used to increase the word accuracy [7].

Two choices have been left open in the technique described above. One is the rather arbitrary scaling $\alpha > 0$ applied to the scores, which does not change the ranking of the hypotheses. For small α, the probability distribution over the hypotheses becomes more uniform, pulling down the confidence of the best path. For large α, the probability mass tends to be concentrated on the best path, yielding higher confidence scores. In [2], $\alpha = 1$ is used, so that the scale of the language model score is one, in [6], $\alpha < 1$. The other choice that needs to be made is how to combine the frame confidences to obtain word-level confidence. We report experiments with the arithmetic average, the geometric average, the minimum and the maximum values. We have evaluated the various choices on our test set A, Broadcast News, using NMCE as a figure of merit. The results are shown in Figure 1. A nearly identical behavior has been observed on test B, Switchboard, and confirmed by ROC plots. The arithmetic average slightly outperforms the geometric average and clearly outperforms the minimum, for an optimal scale factor $\alpha = 0.6$. More surprising is the performance of the maximum value, which works best for a scale factor $\alpha = 0.9$. Taking the maximum frame confidence over the duration of a word tends to yield more words with the maximum confidence score of one, narrow the range of confidence scores. We consider both choices ($\alpha = 0.6$ & average and $\alpha = 0.9$ & max) in the next section.

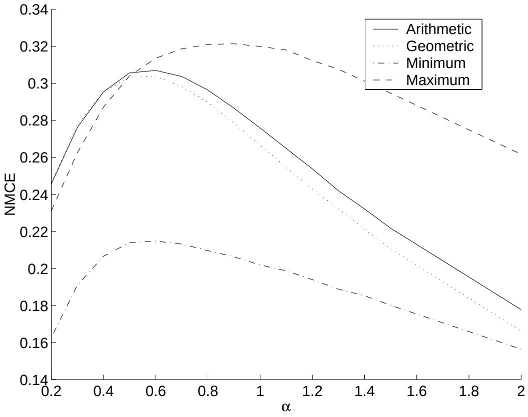

Fig. 1. Effect of scaling and combining technique on performance.

4. LINEAR COMBINATION OF CONFIDENCE MEASURES

We are trying to determine whether combining the lattice-based confidence measure described in section 3 with other techniques can lead to a significant increase on performance, and whether there is a combination that perform well across all tasks. We consider several features that we have found to be useful for confidence annotation:

LB the *lattice-based confidence*, with scale 0.6, arithmetic average,

LX the *lattice-based confidence*, with scale 0.9, maximum over word duration,

AC the *acoustic score* of each word, divided by the word duration,

SC the *search score*, i.e. the weighted sum of acoustic and language model scores, divided by the word duration,

BK the *background difference score*, defined as the difference of between the Viterbi score of the word and the score of a filler model on the same time interval, divided by the word duration.

Set	LB	LX	AC	SC	BK
A	0.307	0.321	0.128	0.165	0.130
B	0.228	0.242	0.087	0.089	0.093
C	0.304	0.298	0.241	0.232	0.248
D	0.216	0.213	0.278	0.278	0.269
E	0.290	0.299	0.270	0.270	0.290
F	0.434	0.462	0.082	0.082	0.090

Table 2. NMCE of several confidence measures used alone

Table 2 shows the performance of each feature used alone. For each test set, linear combinations of all pairs of features are optimized for NMCE. The optimal NMCE for all pairs are reported in Table 3, with the exception of the pairs involving the search score, which do not bring significant gains over pairs involving the acoustic score instead. In addition, the optimal NMCE for 3-feature combinations are reported in the last column. The number in parenthesis in the BK/LB cells is the NMCE of the BK+LB combination *with weights optimized for the all six test sets*, to be compared with the figure in the LB/BK cells.

Since the objective of this experiment is not to report absolute performance values, but to select and compare combinations of features, we feel entitled to optimize the linear weights on the same test for which we report NMCE values. Yet, we have checked that even on the smaller test sets, overfitting does not occur.

Several conclusions can be drawn from those figures. First, alone or in combination, taking the maximum frame confidence over the word duration is slightly better than averaging it. Next, combining the lattice-based confidence with the acoustic score or the background difference score yields a significant gain in most cases. On the other hand, there is little evidence that adding a third feature from those that we considered would be helpful. It is also interesting to note that the same weights can be used across all test sets with little loss of performance compared to weights specifically optimized for each test set. Finally, the automatic optimization of the weights by minimizing the NMCE consistently yield solutions that match those obtained by examining ROC curves.

5. REJECTION OF OUT-OF-GRAMMAR UTTERANCES

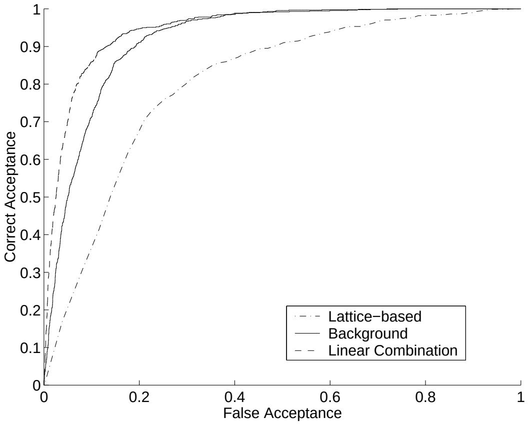

Fig. 2. Receiver Operating Characteristic for rejection of out-of-grammar utterance, car navigation data.

In this section we apply the same techniques as above to the problem of out-of-grammar utterance rejection. Out-of-grammar

Broadcast News					
	LB	LX	AC	BK	
LB	0.307	0.324	0.327	0.328	LB+AC+BK
LX		0.321	0.340	0.341	0.329
AC			0.128	0.130	LX+AC+BK
BK	(0.328)			0.130	0.341

Switchboard					
	LB	LX	AC	BK	
LB	0.228	0.242	0.235	0.234	LB+AC+BK
LX		0.242	0.251	0.248	0.237
AC			0.087	0.095	LX+AC+BK
BK	(0.234)			0.093	0.252

Travel reservations					
	LB	LX	AC	BK	
LB	0.304	0.304	0.362	0.360	LB+AC+BK
LX		0.298	0.363	0.362	0.370
AC			0.241	0.258	LX+AC+BK
BK	(0.353)			0.248	0.368

Car commands					
	LB	LX	AC	BK	
LB	0.216	0.220	0.384	0.381	LB+AC+BK
LX		0.213	0.402	0.390	0.391
AC			0.278	0.285	LX+AC+BK
BK	(0.354)			0.269	0.407

Stock Names					
	LB	LX	AC	BK	
LB	0.290	0.299	0.395	0.426	LB+AC+BK
LX		0.299	0.397	0.414	0.427
AC			0.270	0.290	LX+AC+BK
BK	(0.382)			0.290	0.419

Car navigation					
	LB	LX	AC	BK	
LB	0.434	0.462	0.448	0.446	LB+AC+BK
LX		0.462	0.468	0.469	0.449
AC			0.082	0.091	LX+AC+BK
BK	(0.440)			0.090	0.468

Table 3. NMCE of linear combinations of features for word-level confidence. Left: two components, right: three components, in parenthesis: weights optimized on all test sets.

utterances come in two flavors. One consists of non-speech events, like breath noises, laughter, background noises, music, etc. The other consists of well-formed utterances that do not belong to the set allowed by the grammar. We focus on the second, and more difficult, class.

The data of our last test set is recognized using 14 different grammars (city names, hotels, addresses, etc.). In order to generate out-of grammar samples, we decoded every utterance with all grammars. In an attempt to focus only on the out-of-grammar rejection problem, we discarded all the erroneous decodings obtained with the correct grammar, and all the correct decoding obtained with the wrong grammars (since some of the grammars overlap). During the optimization of the combination weights, a larger weight was given to the in-grammar samples in order to offset the larger amount of out-of-grammar samples. The word-level features used in the previous section are averaged in order to obtain sentence-level features.

	LB	LX	AC	BK	
LB	0.240	0.244	0.577	0.591	LB+AC+BK
LX		0.222	0.577	0.588	0.597
AC			0.502	0.520	LX+AC+BK
BK				0.510	0.594

Table 4. NMCE of pair-wise linear combinations for out-of-grammar utterance rejection, Car navigation.

The effectiveness of the lattice-based measure is less than for word-level confidence annotation. Yet, as in the latter case, its addition to either the acoustic score or the background difference score yields a significant improvement, as shown by the ROC on Figure 2.

6. CONCLUSIONS

We have proposed a new figure of merit, the Normalized Maximum Cross Entropy, that is well suited to word or utterance rejection. We used it as an objective function to choose the parameters of a confidence measure based on word graphs, and to optimize combinations of various confidence measures. It was found that combining the same two features, including the one based on word graphs, performs well on several, very different, tasks. Rejection of out-of-grammar utterances was also addressed using the same techniques.

7. REFERENCES

[1] T. Kemp and T. Schaaf, "Estimating confidence using word lattices," in *Proc. Eurospeech*, Rhodes, Sept. 1997, pp. 827–830.

[2] G. Evermann and P. Woodland, "Large vocabulary decoding and confidence estimation using word posterior probabilities," in *Proc. ICASSP*, Istambul, 2000.

[3] T. Hazen, T. Burianek, J. Polifroni, and S. Seneff, "Recognition confidence scoring for use in speech understanding systems," in *Proc. Automatic Speech Recognition Workshop*, Paris, Sept. 2000, pp. 213–220.

[4] M. Ayer, H. Brunk, G. Ewing, W. Reid, and E. Silverman, "An empirical distribution function for sampling with incomplete information," *Ann. Math. Statist.*, vol. 26, pp. 641–647, 1954.

[5] M. Siu, H. Gish, and F. Richardson, "Improved estimation, evaluation and application of confidence measures for speech recognition," in *Proc. Eurospeech*, Rhodes, Sept. 1997, vol. 2, pp. 831–834.

[6] F. Wessel, K. Macherey, and H. Ney, "A comparison of word graph and N-best list based confidence measures," in *Proc. ICASSP*, Istambul, June 2000, pp. 1587–1590.

[7] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus among words: lattice-based word error minimization," in *Proc. Europseech*, Budapest, 1999, pp. 495–498.