USING WORD PROBABILITIES AS CONFIDENCE MEASURES

Frank Wessel, Klaus Macherey and Ralf Schlüter

Lehrstuhl für Informatik VI, RWTH Aachen – University of Technology
52056 Aachen, Germany
wessel@informatik.rwth-aachen.de

ABSTRACT

Estimates of confidence for the output of a speech recognition system can be used in many practical applications of speech recognition technology. They can be employed for detecting possible errors and can help to avoid undesirable verification turns in automatic inquiry systems. In this paper we propose to estimate the confidence in a hypothesized word as its posterior probability, given all acoustic feature vectors of the speaker utterance. The basic idea of our approach is to estimate the posterior word probabilities as the sum of all word hypothesis probabilities which represent the occurrence of the same word in more or less the same segment of time. The word hypothesis probabilities are approximated by paths in a wordgraph and are computed using a simplified forward-backward algorithm. We present experimental results on the NORTH AMERICAN BUSINESS (NAB'94) and the German VERBMOBIL recognition task.

1. INTRODUCTION

With the rising number of different application areas for speech recognition systems, the demand for the ability to spot erroneous words also increases. Confidence measures can be successfully used for tagging the output of a speech recognizer with either 'correct' or 'incorrect', enabling the recognition system to spot the position of possible errors in its output. In automatic inquiry systems, e.g. train timetable information systems or switchboards, confidence measures can be employed to avoid unnecessary and annoying verification turns if the confidence for the relevant keywords in the speaker utterance is high enough. In this case, no explicit verification is needed and the dialogue duration can be drastically shortened.

Previous work on confidence measures has either investigated the computation of confidence measures during the acoustic decoding process, e.g. [1, 2] or the computation of confidence measures on the basis of word lattices, e.g. [4] and n-best lists, e.g. [6]. Gillick et at. [3] have estimated and evaluated their confidence measure in the framework of a probabilistic approach, making use of generalized linear models for relating a confidence feature vector directly to the probability of a word to be correct. Weintraub et at. [9] have used artifical neural networks to model the relation between the different features and this probability.

The computation of posterior word probabilities in this paper can be seen as an extension of [3], i.e. interpreting confidence as the probability that the recognized word is correct and as an extension of the word graph link probabilities presented in [4] which can be regarded as posterior probabilities for hypotheses in the word-graph. The novel contribution of this paper is the estimation of the posterior word probabilities as the sum of all posterior word hypothesis probabilities which represent the occurrence of the same word in more or less the same segment of time.

This work was partly funded by the European Commision in the framework of the ARISE project under grant LE3-4229. The responsibility for the contents of this study lies with the authors.

2. COMPUTING HYPOTHESIS PROBABILITIES

The posterior word hypothesis probability for a word hypothesis w with starting and ending time t_a and t_e respectively – i.e. starting with feature vector x_{t_a} and ending with feature vector x_{t_e} – given a sequence of acoustic feature vectors x_1^T, is computed in the framework of a forward-backward algorithm, summing up the posterior probabilities of all those word hypothesis sequences which contain the word hypothesis w with the same starting and ending time.

$$p(w, t_a, t_e | x_1^T) = \sum_{W_a} \sum_{W_e} p(W_a, w, W_e | x_1^T)$$

$$= \frac{\sum_{W_a} \sum_{W_e} p(x_1^T | W_a, w, W_e) \cdot p(W_a, w, W_e)}{p(x_1^T)} \quad , \qquad (1)$$

where W_a denotes all word hypothesis sequences preceeding w and W_e all those succeeding w. $p(x_1^T | W_a, w, W_e)$ is the acoustic model probability, $p(W_a, w, W_e)$ the language model probability and

$$p(x_1^T) = \sum_{w} \sum_{W_a} \sum_{W_e} p(x_1^T | W_a, w, W_e) \cdot p(W_a, w, W_e) \quad . \qquad (2)$$

Since a word graph is a compact representation of the most probable word sequences, the summation can be restricted to all word hypothesis sequences represented in the word graph. This simplification can easily be justified, as the probability of the sequences contained in the word graph should clearly dominate the remaining probability mass of word hypothesis sequences not contained in it.

Let us assume that we use a conventional m-gram language model for obtaining the conditional language model probabilities $p(w|h)$, where $h = (h_1 ... h_{m-1})$ is the history of word w. Regarding h as an equivalence class containing all word sequences whose last words are identical to h, we can now compute the 'forward' probability $\Phi_t(h)$. $\Phi_t(h)$ is the probability that the last $m-1$ word hypotheses of a word hypothesis sequence ending at time t

are identical to h:

$$\Phi_t(h) = \sum_{W_a \in h} p(x_1^t|W_a) \cdot p(W_a) \quad . \tag{3}$$

One should bear in mind that the beginning of a word hypothesis sequence requires special treatment because no language model history or only a short history is known. The joint probability of the word hypothesis sequence $W_a = (a_1...a_N)$ is therefore computed as:

$$p(W_a) = p(a_1) \cdot \prod_{i=2}^{m-1} p(a_i|a_1^{i-1}) \cdot \prod_{i=m}^{N} p(a_i|a_{i-m+1}^{i-1}) \quad . \tag{4}$$

In our speech recognition system a word graph [5] is a directed graph whose nodes are interpreted as starting and ending times of word hypotheses and whose edges represent word hypotheses. The acoustic probabilities $p(x_{t_a}^{t_e}|w)$ are therefore stored at the edges. Once the wordgraph is sorted on the starting times of the word hypotheses contained in it, dynamic programming can be applied and the 'forward' probabilities can be computed successively in an ascending order:

$$\Phi_{t_e}\left(h_2^{m-1}, w\right) = p(x_{t_a}^{t_e}|w) \quad \cdot \quad \sum_{h_1} \Phi_{t_a-1}(h_1, h_2^{m-1}) \\ \cdot \quad p(w|h_1, h_2^{m-1}) \quad . \tag{5}$$

Since t_a is the starting time of word w, $t_a - 1$ denotes the ending time of the preceeding word h_{m-1}. Analogously, let $\Psi_t(f)$ denote the 'backward' probability that the first $m-1$ word hypotheses of a word hypothesis sequence beginning at time t are identical to $f = (f_1...f_{m-1})$:

$$\Psi_t(f) = \sum_{W_e \in f} p(x_t^T|W_e) \cdot p(W_e) \quad . \tag{6}$$

As h above, f is interpreted as an equivalence class, this time containing all word hypothesis sequences which start with $(f_1...f_{m-1})$. As the word hypotheses preceeding W_e in Equation (6) are not known, $p(W_e)$ must be interpreted as in Equation (4). On the other hand, the prediction of the word hypotheses contained in f must be based on the full history length. We therefore compute a 'modified backward' probability:

$$\widetilde{\Psi}_t(f) = \sum_{W_e \in f} p(x_t^T|W_e) \cdot \prod_{i=m}^{M} p(e_i|e_{i-m+1}^{i-1}) \quad , \tag{7}$$

where $W_e = (e_1...e_M)$. The missing language model probabilities are included later on when computing the posterior word hypothesis probabilities. Equation (7) can be evaluated using dynamic programming as well. The word graph is sorted on the ending times of the word hypotheses and the 'modified backward' probabilities are computed in a descending order:

$$\widetilde{\Psi}_{t_a}\left(w, f_1^{m-2}\right) = p(x_{t_a}^{t_e}|w) \quad \cdot \quad \sum_{f_{m-1}} \widetilde{\Psi}_{t_e+1}(f_1^{m-2}, f_{m-1}) \\ \cdot \quad p(f_{m-1}|w, f_1^{m-2}) \quad . \tag{8}$$

With the definitions in Equations (1), (5) and (8), the posterior word hypothesis probability can now be computed as follows:

$$\begin{aligned} & p(w, t_a, t_e|x_1^T) \\ & = \sum_{h_2^{m-1}} \sum_{f_1^{m-2}} \frac{\Phi_{t_e}(h_2^{m-1}, w) \cdot \widetilde{\Psi}_{t_a}(w, f_1^{m-2})}{p(x_1^T) \cdot p(x_{t_a}^{t_e}|w)} \\ & \cdot \prod_{i=1}^{m-2} p(f_i|h_{i+1}^{m-1}, w, f_1^{i-1}) \quad . \end{aligned} \tag{9}$$

$p(x_1^T)$ in the denominator can be evaluated as follows:

$$\begin{aligned} p(x_1^T) &= \sum_{h_2^{m-1}} \sum_{w} \Phi_T(h_2^{m-1}, w) \\ &= \sum_{f_1^{m-2}} \sum_{w} \widetilde{\Psi}_1(w, f_1^{m-2}) \\ & \quad \cdot \left[\prod_{i=1}^{m-2} p(f_i|w, f_1^{i-1})\right] \cdot p(w) \quad . \end{aligned} \tag{10}$$

The last term in Equation (9) represents the language model probabilities which are missing in Equation (7), as mentioned above.

Usually, the language model scores are multiplied with a language model scaling factor. During the recognition or rescoring phase this strategy is equivalent to scaling the acoustic score down (with the reciprocal language model scaling factor). When computing posterior probabilities as specified in Equation (9), these two approaches are no longer equivalent. Besides numerical problems which we have noticed when using the language model scaling factor, the sums in Equation (9) are dominated by only a few word graph hypotheses, because of the large differences in the acoustic scores. In our opinion, these differences are mainly due to the variance of the acoustic features which is generally underestimated. If a reestimation of these variances is not feasible, the acoustic scores should at least be scaled down in order to obtain a useful result. The acoustic scaling factors have been estimated on the cross-validation corpora beforehand.

3. COMPUTING WORD PROBABILITIES

The posterior word hypothesis probability defined in Equation (9) has shown to have a very poor discriminating ability between 'correct' and 'false' words in all of our preliminary experiments. Actually, this result is not surprising when considering the fact that the fixed starting and ending time of a word hypothesis in the word graph are more or less arbitrary. In fact, the posterior probabilities of all those word graph hypotheses annotated with the word index of the current word hypothesis for which we try to compute a measure of confidence should be added if the starting and ending times of these hypotheses only slightly differ from those of the word hypothesis under consideration.

A naive approach to specifying the vague definition of 'slightly differing' starting and ending times would be to experiment with different percentages of overlaps between the current word hypothesis and all other hypotheses with the same word index. In fact, this approach can be successfully used and there is no effect on the confidence error rate defined in section 4, as long as the minimal overlap ranges between 0% and 30%. Overlaps above 30% have

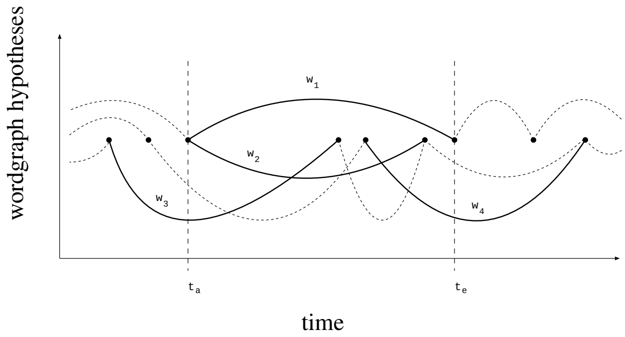

Figure 1: The plain lines indicate different hypotheses for the same word w, the dotted lines represent hypotheses for other words.

produced worse results. It is therefore intuitive to simply sum up all posterior probabilities of those hypotheses with the same word index as long as they have an overlap with the current hypothesis at all.

The problem with this definition is that we no longer have a probability distribution. The posterior word hypothesis probabilities sum up to unity for each time frame by definition, but since the posterior word probabilities as defined above are not added up over only one time frame, we run into problems. As shown in Figure 1, the word hypotheses w_3 and w_4 do not overlap. Still, when computing the accumulated posterior word probability for hypothesis w_1, both probabilities would be included in the sum. To avoid this problem, the summation of posterior hypotheses must be carried out on the basis of time frames. Therefore, we have added the posterior probabilities of all hypotheses for the same word for each time frame between t_a and t_e which they intersect and estimated the posterior word probabilites as the arithmetic mean over all these time frames or as the maximum, respectively. We have detected no difference between these two criteria and have decided to use the maximum for algorithmic reasons. We can now guarantee that the word posterior probabilities sum up to unity. The posterior word probability which we use as a measure of confidence is therefore defined as:

$$\begin{aligned} &\widetilde{p}(w, t_a, t_e | x_1^T) \\ &= \max_{t: t_a \leq t \leq t_e} \sum_{(t_i, t_j): t_i \leq t \leq t_j} p(w, t_i, t_j | x_1^T) \end{aligned} \tag{11}$$

where t_i and t_j are the starting and ending time of the hypothesized word w. We have used this criterion in the following evaluation experiments. It can directly be interpreted as the probability that the word under consideration is correct.

4. EXPERIMENTAL RESULTS

We have decided to evaluate our confidence measure using the confidence error rate (CER) which is simply defined as the number of incorrectly assigned tags divided by the total number of recognized words. Another criterion is the normalized cross entropy S as proposed by NIST. In our opinion, this quantity is not useful for evaluating our confidence measure. In order to be able to discuss the disadvantage of this criterion, we first give a definition:

$$S = \frac{H(C) - H(C|X)}{H(C)} \quad . \tag{12}$$

$H(C)$ is the initial entropy of the recognizer output when tagging all words with 'correct' and $H(C|X)$ can be interpreted as the entropy of the tag sequence attached to the recognizer output, provided with the information contained in the confidence measure. For details, the reader is referred to [8]. Although S can easily be interpreted as the relative reduction in entropy, it is no longer sensibly defined as soon as the posterior probability for a word to be correct equals one, even though the word has not been recognized correctly. If this happens only once in the testing corpus, it will have almost no effect on the overall quality of the confidence measure. Still, the normalized cross entropy will approach infinity. One way to elude this problem is to guarantee that the posterior probabilities never equal one, either by removing all words from the test corpus whose posterior probabilities are identical to one or by inserting alternative hypotheses into the word graph. Both methods circumvent the restrictions imposed by the recognition task itself. With a given wordgraph and no alternative to a word hypothesis one cannot do anything but state that this word has been recognized correctly. Moreover, the confidence error rate is more intuitive and practically oriented. We therefore confine ourselves to the use of this quantity.

We have performed evaluation experiments on two different corpora, on the North American Business corpus (NAB'94 H1 development corpus) and on the offical evaluation corpus of the 1996 VERBMOBIL recognition task. The VERBMOBIL translation system generates speech-to-speech translations between German and English in the appointment scheduling domain. The corpus consists of spontaneous human-to-human dialogs, including noises, hesitations and false starts.

As Siu et at. [8] have pointed out, the improvements obtained with confidence measures are very sensitive to the recognition operating points across different recognition systems and can easily be increased by changing the baseline confidence error rate which is identical to the number of correctly recognized words divided by the total number of recognized words. We therefore give a detailed specification of the word graphs used and the baseline recognition results which we have obtained with our word graphs in Table 1. For the definition of the word graph density (WGD), the node graph density (NGD), the boundary graph density (BGD) and for the graph error rate (GER) and for further details on the specification of word graphs, the reader is referred to [5]. Figure 2 shows the probability histogram for the two classes 'correct' and 'false'.

Table 1: Wordgraph specification for the NAB'94 H1 and the VERBMOBIL corpus

corpus	spoken words	WGD	NGD	BGD	GER [%]
NAB'94 H1					
cross-val.	8186	352.8	54.5	10.9	5.1
eval.	7387	1174.9	156.3	18.6	4.2
VERBMOBIL					
cross-val.	11129	308.8	47.3	12.0	8.2
eval.	5421	328.6	52.8	12.4	6.7

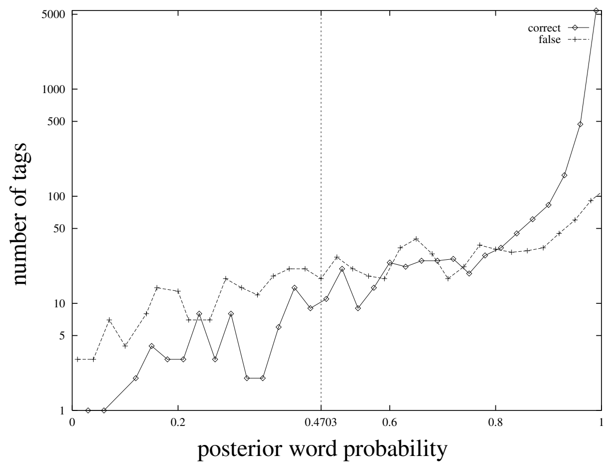

Figure 2: Histogram plot of the posterior word probabilities for the two classes 'correct' and 'false' on the NAB'94 H1 development corpus using a trigram language model.

The number of events with the same probability is plotted on a logarithmic scale. The decision whether to tag a word as 'correct' or false' is based on thresholding the posterior probability of this word as defined in Equation (11). The threshold is optimized on the cross-validation corpora beforehand and is about 0.47 in this specific case. Words with a smaller posterior probability than the threshold are tagged as false and all other words as correct. The results we have obtained are summarized in Table 2. The baseline confidence error rate is identical to the number of insertions and substitutions divided by the total number of recognized words. Our results on the NAB'94 H1 recognition task are promising. In terms of the relative improvement in the confidence error rate, we have noticed almost no difference when using a trigram instead of a bigram language model. With only the posterior word probability as a confidence measure, we have obtained an improvement of 21% - 23% relatively. The relative reduction of the confidence error rate on the VERBMOBIL task is lower for both a bigram and a trigram language model. Still, with only one feature we have obtained a reduction of about 14% - 18% relatively. It is interesting to note that the probability thresholds which have been adjusted on the cross-validation corpora are almost optimal for the evaluation corpora.

Table 2: Results for the confidence measure on the two evaluation corpora

corpus	errors [%] del/ins/WER	baseline CER [%]	CER [%]
NAB'94 H1			
bigram:	2.4/2.6/16.3	13.9	10.7
trigram:	1.7/2.5/13.7	11.9	9.4
VERBMOBIL			
bigram:	4.4/3.9/21.7	17.4	14.9
trigram:	3.8/3.2/19.4	15.7	12.9

5. CONCLUSION

We have proposed to use a posterior word probability as a confidence measure for a word in the output of a speech recognizer. We have used a forward-backward algorithm in the word graph to compute posterior hypothesis probabilities and estimated the posterior word probability summing up these probabilities over one time frame between the starting and ending time of the word hypothesis under consideration. We have discussed the restricted applicability of the normalized cross entropy for evaluating confidence measures and we have presented results on the NAB'94 H1 and the German VERBMOBIL recognition task for a bigram and trigram language model. We have obtained relative improvements in the confidence error rate between 14% and 23%.

6. REFERENCES

[1] L. Chase: 'Word and Acoustic Confidence Annotation for Large Vocabulary Speech Recogniton', in Fifth Europ. Conf. on Speech Communication and Technology, Rhodes, Greece, pp. 815-818, September 1997.

[2] S. Cox, R. Rose: 'Confidence Measures for the Switchboard Database', in Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing 1996, Atlanta, USA, pp. 511-514, May 1996.

[3] L. Gillick, Y. Ito, J. Young: 'A Probabilistic Approach to Confidence Measure Estimation and Evaluation', in Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing 1997, Munich, Germany, pp. 879-882, April 1997.

[4] T. Kemp, T. Schaaf: 'Estimating Confidence Using Word Lattices', in Fifth Europ. Conf. on Speech Communication and Technology, Rhodes, Greece, pp. 827-830, September 1997.

[5] S. Ortmanns, H. Ney, X. Aubert: 'A Word Graph Algorithm for Large Vocabulary Continuous Speech Recognition', Computer, Speech and Language, vol. 11, no. 1, pp. 43-72, January 1997.

[6] Bernhard Rueber: 'Obtaining confidence measures from sentence probabilities', in Fifth Europ. Conf. on Speech Communication and Technology, Rhodes, Greece, pp. 739-742, September 1997.

[7] T. Schaaf, T. Kemp: 'Confidence Measures For Spontaneous Speech Recognition', in Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing 1997, Munich, Germany, pp. 875-878, April 1997.

[8] M. Siu, H. Gish and F. Richardson: 'Improved Estimation, Evaluation and Applications of Confidence Measures for Speech Recognition', in Fifth Europ. Conf. on Speech Communication and Technology, Rhodes, Greece, pp. 831-834, September 1997.

[9] M. Weintraub, F. Beaufays, Z. Rivlin, Y. Konig, A. Stolcke: 'Neural-Network Based Measures of Confidence for Word Recognition', in Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing 1997, Munich, Germany, pp. 887-890, April 1997.