

# A COMPARISON OF WORD GRAPH AND N-BEST LIST BASED CONFIDENCE MEASURES

Frank Wessel, Klaus Macherey, and Hermann Ney

Lehrstuhl für Informatik VI, RWTH Aachen – University of Technology  
Ahornstraße 55, 52056 Aachen, Germany  
wessel@informatik.rwth-aachen.de

## ABSTRACT

In this paper we present and compare several confidence measures for large vocabulary continuous speech recognition. We show that posterior word probabilities computed on word graphs and N-best lists clearly outperform non-probabilistic confidence measures, e.g. the acoustic stability and the hypothesis density. In addition, we prove that the estimation of posterior word probabilities on word graphs yields better results than their estimation on N-best lists and discuss both methods in detail. We present experimental results on three different corpora, the English NAB '94 20k development corpus, the German VERBMOBIL '96 evaluation corpus and a Dutch corpus, which has been recorded with a train timetable information system in the ARISE project.

## 1. INTRODUCTION

In previous studies, the combination of several confidence features was investigated. These features were collected during the acoustic decoding process, e.g. [1] or were extracted from N-best lists and word graphs, e.g. [2, 5]. Most of these features were non-probabilistic and had to be combined to form a single confidence measure.

In this paper we estimate the confidence in a hypothesized word following a probabilistic approach. We associate the confidence in a word directly with its posterior probability.

In [6] we computed these posterior probabilities on word graphs using a modification of the forward-backward algorithm. Usually, a word is hypothesized more than once and the word graph thus contains several hypotheses for the same word with different starting and ending times (later referred to as *segmentation*). In order to obtain satisfying results we had to sum up the posterior probabilities of these hypotheses for each word in the reference sentence.

We now extend our previous work in several directions. First, we study three different methods of determining which hypothesis probabilities have to be accumulated. Second, we estimate the posterior probabilities on N-best lists instead of word graphs, in an approach comparable to the work presented in [5]. The main advantage of N-best lists over word graphs is that the computation of posterior probabilities can directly be carried out on the basis of word positions. The accumulation of posterior probabilities might thus be unnecessary. Third, we compare both approaches with alternative criteria, i.e. the acoustic stability [4] and the hypothesis density [2].

## 2. WORD PROBABILITIES ON WORD GRAPHS

In the following, we briefly describe the most important aspects of our previous work. For details, the reader is referred to [6]. The posterior probability for a word hypothesis  $(t_s, t_e, w)$  with starting time  $t_s$  and ending time  $t_e$ , given a sequence of acoustic feature vectors  $x_1^T = x_1 x_2 \dots x_T$ , is computed with a forward-backward algorithm. We sum up the posterior probabilities of all

those sentences which contain this specific word hypothesis:

$$\begin{aligned} p_{t_s, t_e}(w|x_1^T) &= \sum_{W_s} \sum_{W_e} p(W_s w W_e | x_1^T) = \\ &= \frac{1}{p(x_1^T)} \cdot \sum_{W_s} \sum_{W_e} \{ p(x_1^{t_s-1} | W_s) \cdot p(x_{t_s}^{t_e} | w) \cdot \\ &\quad \cdot p(x_{t_e+1}^T | W_e) \cdot p(W_s w W_e) \} \quad , \end{aligned} \quad (1)$$

where  $W_s$  denotes all word sequences preceding  $w$  and  $W_e$  all those succeeding  $w$ .

Since a word graph is a compact representation of the most probable sentences, the summation can be restricted to all hypothesis sequences contained in it. Let us assume that we use an m-gram language model and let  $h_1^{m-1} = h_1 h_2 \dots h_{m-1}$  be the history of word  $w$ . Regarding the word history as an equivalence class containing all word sequences whose last words are  $h_1^{m-1}$ , we can now compute the forward probability that the last words of a word sequence  $W_s = w_1 w_2 \dots w_{|W_s|}$  ending at time  $t$  are identical to  $h_1^{m-1}$ :

$$\Phi_t(h_1^{m-1}) = \sum_{W_s \in h_1^{m-1}} p(x_1^t | W_s) \cdot \prod_{n=1}^{|W_s|} p(w_n | w_{n-1}^{m-1}) \quad , \quad (2)$$

where  $|W_s|$  denotes the number of words in  $W_s$ . This equation can be solved recursively:

$$\begin{aligned} \Phi_t(h_2^{m-1} w) &= p(x_{t_s(w)}^t | w) \cdot \\ &\quad \cdot \sum_{h_1} \Phi_{t_s(w)-1}(h_1 h_2^{m-1}) \cdot p(w | h_1 h_2^{m-1}) \quad . \end{aligned} \quad (3)$$

Since  $t_s(w)$  is the starting time of word  $w$ ,  $t_s(w) - 1$  denotes the ending time of the preceding word  $h_{m-1}$ . Analogously,  $\tilde{\Psi}_t(f_1^{m-1})$  denotes the backward probability that the first word hypotheses of a word sequence  $W_e = w_1 w_2 \dots w_{|W_e|}$  beginning at time  $t$  are identical to  $f_1^{m-1} = f_1 f_2 \dots f_{m-1}$ :

$$\tilde{\Psi}_t(f_1^{m-1}) = \sum_{W_e \in f_1^{m-1}} p(x_t^T | W_e) \cdot \prod_{n=m}^{|W_e|} p(w_n | w_{n-m+1}^{m-1}) \quad . \quad (4)$$

$|W_e|$  denotes the number of words in  $W_e$ . The missing language model probabilities for all words  $w_1 w_2 \dots w_{m-1}$  are computed later in Equation (6) for algorithmic reasons.

Equation (4) can be evaluated recursively as well. The word graph is sorted on the ending times of the word hypotheses and the backward probabilities are computed in a descending order:

$$\begin{aligned} \tilde{\Psi}_t(w f_1^{m-2}) &= p(x_t^{t_e(w)} | w) \cdot \\ &\quad \cdot \sum_{f_{m-1}} \tilde{\Psi}_{t_e(w)+1}(f_1^{m-2} f_{m-1}) \cdot p(f_{m-1} | w f_1^{m-2}) \quad . \end{aligned} \quad (5)$$

With the definitions in Equations (1), (3) and (5), the posterior hypothesis probability can now be computed as follows:

$$\begin{aligned}
p_{t_s, t_e}(w|x_1^T) &= \\
&= \sum_{h_2^{m-1}} \sum_{f_1^{m-2}} \frac{\Phi_{t_e}(h_2^{m-1} w) \cdot \tilde{\Psi}_{t_s}(w f_1^{m-2})}{p(x_1^T) \cdot p(x_{t_s}^{t_e}|w)} \cdot \\
&\quad \cdot \prod_{n=1}^{m-2} p(f_n|h_{n+1}^{m-1} w f_1^{n-1}) \quad . \quad (6)
\end{aligned}$$

In addition to a language model scaling factor, we also use a scaling factor  $\alpha < 1$  to scale down both, the acoustic and the language model probabilities. This scaling has a major impact on the performance of the posterior probabilities. If the scores are not scaled appropriately, the sums in all of the equations above are dominated by only a few word graph hypotheses because of the large differences in the acoustic scores. For details see [6].

The posterior hypothesis probability defined in Equation (6) can now be used as a measure of confidence in the word under consideration:

$$\mathcal{C}(w) := p_{t_s, t_e}(w|x_1^T) \quad . \quad (7)$$

In the experiments, the confidence is compared with a tagging threshold, which has to be optimized on a cross validation corpus beforehand. Words whose confidence exceeds this threshold are tagged as ‘correct’, all others as ‘false’.

Equation (7) only has a very poor discriminating ability between these two classes, see Table 3. This observation is not surprising when considering the fact that the fixed starting and ending times of a word hypothesis determine which paths in the word graph are included during the computation of the forward-backward probabilities. The segmentation of the word graph thus has a strong impact on the posterior probabilities of the hypotheses contained in it. Although several hypotheses represent the same word, the probability mass of the word is split among them. The unsatisfactory performance of the confidence measure defined in Equation (7) is thus a strong indication that the segmentation of the word graph must be relaxed.

In the following we describe several relaxation methods we implemented. The underlying principle is to sum up the posterior probabilities of all hypotheses for the same word  $w$  over a common time frame. Since a word usually extends over several time frames, the determination of this time frame has an immediate influence on the selection of hypotheses, whose posterior probabilities are accumulated. In a first attempt we summed up the probabilities of all hypotheses with an identical word index for which the intersection of the time intervals defined by the starting and ending times of the considered hypotheses is not empty:

$$\mathcal{C}_{\square}(w) := \sum_{\substack{(t_s, t_e): \\ [t_s, \dots, t_e] \cap T(w) \neq \emptyset}} p_{t_s, t_e}(w|x_1^T) \quad , \quad (8)$$

where  $T(w) = [t_s(w) \dots t_e(w)]$  denotes the time interval defined by the starting and ending times of the hypothesis for word  $w$ . As the results in Table 3 show, Equation (8) performs significantly better than Equation (7) on all testing corpora.

Unfortunately, the sum of the accumulated posterior probabilities over all different words for one time frame does no longer sum up to unity by definition. Although performing better in terms of confidence error rate (defined in [2, 6]) the question remains, whether the missing normalization has an effect on the confidence measure. If we restrict the accumulation of the posterior hypothesis probabilities to a single time frame, common to

all hypotheses for a specific word, the accumulated probabilities do sum up to unity. Thus, we have implemented the following confidence measure:

$$\mathcal{C}_{\perp}(w) := \sum_{\substack{(t_s, t_e): \\ t \in [t_s, \dots, t_e]}} p_{t_s, t_e}(w|x_1^T) \quad , \quad (9)$$

$$\text{where } t = \left\lceil \frac{t_e(w) - t_s(w)}{2} \right\rceil + t_s(w) \quad .$$

We accumulated the posterior probabilities of all hypotheses for word  $w$  which intersect the middle time frame of the hypothesis under consideration. As our results show, the performance is comparable with the confidence measure defined in Equation (8). Still, the choice of the time frame might be sub-optimal. We therefore carried out the accumulation not only for the middle time frame of the current hypothesis but for all of its time frames and chose the maximum of these values as a measure of confidence:

$$\mathcal{C}_{max}(w) := \max_{t \in T(w)} \sum_{\substack{(t_s, t_e): \\ t \in [t_s, \dots, t_e]}} p_{t_s, t_e}(w|x_1^T) \quad . \quad (10)$$

As the experiments presented in Table 3 indicate, this measure performs slightly, but not significantly better than the one defined in Equation (9).

### 3. WORD PROBABILITIES ON N-BEST LISTS

As described in the previous section, the relaxation of the word graph segmentation is crucial for a reliable estimation of the confidence. One of the main advantages of N-best lists over word graphs is that the different sentence hypotheses contained in the N-best list are based on word positions. In [5] the author suggests to compute posterior probabilities for semantic items in a recognized sentence on N-best lists. This approach is very similar to the computation of posterior probabilities on word graphs and can easily be extended to the computation of posterior probabilities for individual words. In doing so, the segmentation problem can be solved.

Let  $\mathcal{A}$  be the vocabulary of the recognizer. For simplification we define a function  $\mathcal{L}$  which computes the Levenshtein alignment between two sentences  $W \in \mathcal{A}^*$  and  $V \in \mathcal{A}^*$  contained in the N-best list and which returns that word  $v$  in sentence  $V$  which was aligned to the  $n$ -th word in sentence  $W$ .

$$\begin{aligned}
\mathcal{L} : \mathcal{A}^* \times \mathcal{A}^* \times \mathbb{N} &\longrightarrow \mathcal{A} \\
\mathcal{L}(W, V, n) &\longmapsto v \quad . \quad (11)
\end{aligned}$$

Using this definition, the posterior word probability for each word  $w_n$  in sentence  $W$  can be computed on N-best lists:

$$\begin{aligned}
p_n(w|x_1^T) &= \sum_{V: \mathcal{L}(W, V, n) = w} p(V|x_1^T) = \\
&= \frac{\sum_{V: \mathcal{L}(W, V, n) = w} p(x_1^T|V) \cdot p(V)}{\sum_V p(x_1^T|V) \cdot p(V)} \quad . \quad (12)
\end{aligned}$$

The summation is carried out over all sentences  $V$  contained in the N-best list. As shown later, the computation on N-best list performs worse than the computation on word graphs.

Table 1: Specification of the word graphs.

corpus	WGD	NGD	BGD	GER [%]
VERBMOBIL	131.6	42.6	12.4	4.7
NAB	75.0	35.7	8.5	5.1
ARISE	86.1	37.6	16.3	7.8

#### 4. ALTERNATIVE CRITERIA

In order to compare the posterior probabilities with alternative criteria we implemented the acoustic stability criterion [4] and the hypothesis density criterion [2].

The motivation for the acoustic stability is that a word is most probably correct if it is contained at the same position, specified by the Levenshtein alignment, in the majority of sentences generated with different weighting between the acoustic and the language model scores. In a first step, we rescore the word graph with the standard language model scaling factor  $\alpha_{ref}$  in order to obtain the first-best sentence  $W$ . Second, we rescore the word graph with  $M$  different language model scaling factors and obtain  $M$  alternative first-best sentences  $V_1 \dots V_M$ . All of these sentences are then aligned with the reference sentence  $W$ . The relative frequency of any word taken from the reference sentence at the same position in all of the sentences  $V_1 \dots V_M$  is a direct measure for the acoustic stability:

$$\mathcal{C}_{acu}(w_n) := \frac{1}{M} \cdot \sum_{i=1}^M \delta(w_n, \mathcal{L}(W, V_i, n)) \quad (13)$$

Another criterion suggested previously, is the hypothesis density [2]. In order to reduce the computational complexity during the decoding process, unlikely hypotheses are usually pruned. If a large number of hypotheses have similar scores for a given time frame, no effective pruning will take place and the number of hypotheses in the word graph will be above average. Since a word is usually hypothesized several times with different starting and ending times, we count each word only once while computing the hypothesis density for a given time frame. Each hypothesis is determined by the word index  $w$ , its starting time  $t_s$  and its ending time  $t_e$ . Let  $WG = \{(t_s, t_e, w)\}$  denote the set of all hypotheses contained in the word graph. The hypothesis density for time frame  $t$  can then be computed as follows:

$$D(t) := |\{w : (t_s, t_e, w) \in WG \wedge t \in T(w)\}| \quad (14)$$

To capture the dynamics of this quantity we used the average hypothesis density over  $T(w)$  as our measure of confidence:

$$\mathcal{C}_{den}(w) := \frac{1}{t_e(w) - t_s(w) + 1} \cdot \sum_{t \in T(w)} D(t) \quad (15)$$

#### 5. EXPERIMENTAL RESULTS

We present experimental results on three different corpora. The NAB '94 20k corpus consists of read newspaper articles (vocabulary size 19987 words), recorded under high-quality conditions, the VERBMOBIL '96 corpus of spontaneous human-to-human dialogues (vocabulary size 5532 words), also recorded under high-quality conditions and the ARISE corpus of human-to-machine dialogues (vocabulary size 985 words), recorded over the telephone with an automatic train timetable information system.

Table 1 specifies the word graphs generated with our speech recognition system. For a definition of the word graph density

Table 2: Word error rates on the different testing corpora.

corpus	bigram LM		trigram LM	
	del/ins/WER [%]		del/ins/WER [%]	
VERBMOBIL	4.4/3.9/21.7	3.6/3.4/19.5		
NAB	2.9/2.5/17.4	2.1/2.3/14.7		
ARISE	1.8/3.8/16.5	2.0/3.2/15.9		

Table 3: Confidence error rates in [%] for posterior word probabilities computed on a word graph with different relaxation strategies.

corpus	baseline	$\mathcal{C}$	$\mathcal{C}_{\square}$	$\mathcal{C}_{\perp}$	$\mathcal{C}_{max}$
VERBMOBIL					
bigram LM:	17.4	16.9	14.9	15.0	<b>14.9</b>
trigram LM:	15.9	14.6	13.0	13.2	<b>12.9</b>
NAB					
bigram LM:	14.6	13.3	11.9	11.8	<b>11.8</b>
trigram LM:	12.6	11.7	9.9	9.9	<b>9.9</b>
ARISE					
bigram LM:	14.4	11.8	9.8	9.8	<b>9.7</b>
trigram LM:	13.8	11.8	9.1	9.3	<b>9.1</b>

(WGD), the node graph density (NGD), the boundary graph density (BGD) and the graph error rate (GER), see [3]. Table 2 comprises the baseline word error rates on the three different testing sets.

For all of the following experiments we have optimized all model parameters, i.e. the acoustic scaling factors, the language model scaling factors and the tagging thresholds on a separate cross-validation set beforehand.

Table 3 summarizes the effect of different relaxation strategies used during the computation of posterior word probabilities on the word graphs. As the table clearly indicates, the relaxation of the word graph segmentation is essential for the computation of the confidence measure. Compared to the posterior probability defined in Equation (7) all of the relaxed probabilities perform significantly better. The missing probability normalization for  $\mathcal{C}_{\square}(w)$  seems to have only a minor influence on the performance. Since  $\mathcal{C}_{max}(w)$  yields the best results we chose this criterion as our standard confidence measure for all further comparisons. In Figure 1, we plotted the detection error trade-off curves for the three corpora using Equation (10) and a trigram language model.

Figure 2 clearly shows that the computation of posterior prob-

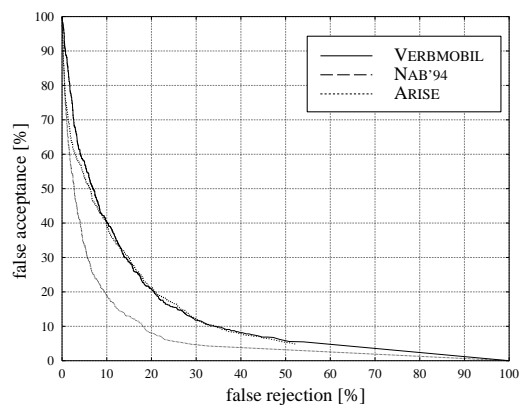


Figure 1: Detection error trade-off for all corpora using a trigram language model.

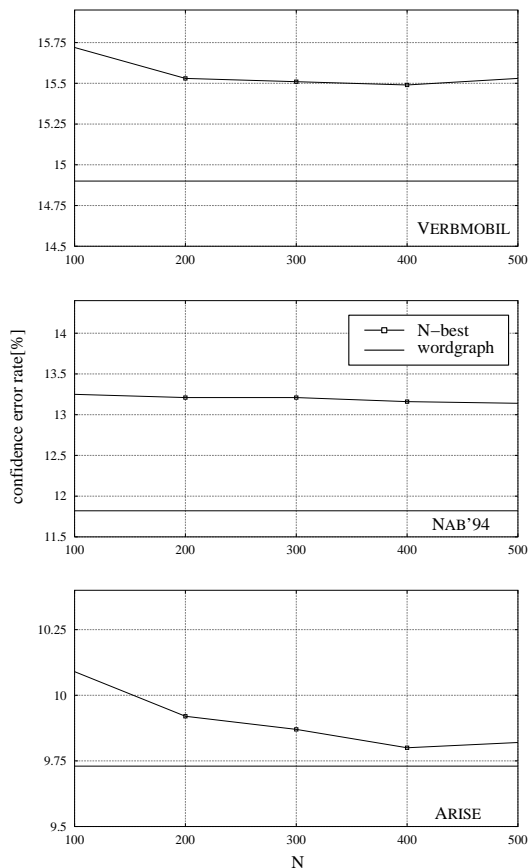


Figure 2: Word graph vs. N-best list for the evaluation corpora using a bigram language model.

abilities on word graphs performs significantly better than the computation on N-best lists, even for rather large values of  $N$ . The effect of the computation on a word basis, which is an advantage over word graphs, is outweighed by several other effects. First, the word graphs represent a drastically larger number of alternative sentences. With an average word graph density of 131.6 and an average sentence length of 18 words on the VERBMOBIL corpus, an upper bound to the mean number of sentences represented by the word graph is  $1.4 \cdot 10^{38}$ , whereas the N-best lists contain only 100 to 500 sentences. Second, the computation on a word basis itself might in fact cause problems. The slight increase in the confidence error rate for  $N = 500$  can be regarded as an indication for this assumption. The information about starting and ending times which is contained in the word graphs and which highly influences the computations carried out in the forward-backward algorithm is in fact not a problem, but very important for a reliable estimation of the posterior probabilities. The rather good performance of the N-best list criterion on the ARISE corpus is caused by the average sentence length of just over three words. The difference between the number of sentences contained in the word graph on the one hand and the N-best list on the other is smaller by orders of magnitude compared to the other two corpora.

The  $M$  language model scales  $\alpha_1 \dots \alpha_M$  chosen for the acoustic stability criterion are equidistant values taken from the interval  $[(1 - \delta) \cdot \alpha_{ref} \dots (1 + \delta) \cdot \alpha_{ref}]$ . For the experiments we used  $\delta = 0.9$  and  $M = 100$ . We noticed only a negligible change in confidence error rate for larger values of  $M$ . As Table 4 indicates, the acoustic stability achieves good results on all corpora. On the other hand, the acoustic stability is clearly not able to outperform the accumulated posterior probability. In addition, the computing time is several times higher than the time

Table 4: Confidence error rates in [%] for the acoustic stability, the hypothesis density and the accumulated posterior probability.

corpus	baseline	$\mathcal{C}_{acu}$	$\mathcal{C}_{den}$	$\mathcal{C}_{max}$
VERBMOBIL				
bigram LM:	17.4	16.4	16.6	<b>14.9</b>
trigram LM:	15.9	14.5	15.4	<b>12.9</b>
NAB				
bigram LM:	14.6	13.0	14.1	<b>11.8</b>
trigram LM:	12.6	10.8	12.5	<b>9.9</b>
ARISE				
bigram LM:	14.4	9.9	10.6	<b>9.7</b>
trigram LM:	13.8	8.9	10.4	<b>9.1</b>

needed for the computation of the accumulated posterior probabilities.

The hypothesis density criterion is also not able to excel the performance of the accumulated posterior probability which is also listed in Table 4 for comparison. The number of parallel hypotheses for a given time frame is not sufficient to be used as a measure of confidence. It is interesting to note that the number of hypotheses for a time frame is implicitly considered in the forward-backward algorithm, anyway.

## 6. CONCLUSION

We presented and compared several confidence measures, based on word graphs and N-best lists. We showed that posterior word probabilities clearly outperform alternative confidence measures, e.g. the acoustic stability and the hypothesis density. In addition, we proved that the estimation of posterior word probabilities on word graphs yields better results than the estimation on N-best lists. The relative reduction in confidence error rate ranges between 18.9% and 34.1% using a trigram language model on different corpora and our best confidence measure, defined in Equation (10).

## 7. REFERENCES

- [1] L. Chase: 'Word and Acoustic Confidence Annotation for Large Vocabulary Speech Recognition', in Fifth Europ. Conf. on Speech Communication and Technology, Rhodes, Greece, pp. 815-818, September 1997.
- [2] 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.
- [3] S. Ortman, 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.
- [4] M. Finke, T. Zeppenfeld, M. Maier, L. Mayfield, K. Ries, P. Zhan, J. Lafferty, A. Waibel: 'Switchboard April 1996 Evaluation Report', Tech. Report Interactive Systems Laboratories, ILKD, April 1996.
- [5] Bernhard Rueber: 'Obtaining Confidence Measures from Sentence Probabilities', in Fifth Europ. Conf. on Speech Communication and Technology, Rhodes, Greece, pp. 739-742, September 1997.
- [6] F. Wessel, K. Macherey, R. Schlüter: 'Using Word Probabilities as Confidence Measures', in Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing 1998, Seattle, USA, pp. 225-228, May 1998.