An Efficient Word Confidence Measure Using Likelihood Ratio Scores

Arlindo O. Veiga*, Fernando S. Perdigão*†

*Institute of Telecommunications, Pole of Coimbra, *†Electrical and Computer Engineering Department
Pole II of University of Coimbra, 3030-290 Coimbra, Portugal
{aveiga, fp}@co.it.pt

Abstract

This paper describes an efficient method to perform word confidence measures in an automatic speech recognition system. The confidence measure is computed during the decoding phase and is based on likelihood ratios between the top hypotheses that reach a word node. Experiments were carried out on a digit database with a connected-digit recognizer. The results show that this method outperforms word-graph confidence measure with a special grammar and is worse with a word loop grammar. Because the proposed confidence measure is evaluated with only one pass, it is very efficient and can be applied with advantage in small or medium vocabulary recognizers, with low computational resources.

1. Introduction

In a speech recognition system there are always recognition errors. A confidence measure of the recognition output becomes an important component of any speech recognition system. Confidence measures can be used for spotting and reject possible errors present in the recognizer output as well as to detect out-of-vocabulary words.

It is well known and reported that the word log-likelihood score itself is not a good indicator of the correctness of the recognized word (e.g. Willett et al. 1998). Most of the studies in this area use N-best lists or word-graphs (Kemp and Schaaf, 1997; Tan et al., 2001; Wessel et al. 98), in order to extract independent confidence information. In some works, a combination of confidence features is used, as well as different classifiers. Features based on the acoustic model, the decoding process, the language model or word semantics are common.

For isolated word recognition systems, the likelihood ratio between the best and the second best hypothesis is a very good confidence estimator, as reported in (Ramalingam et al, 1998). However, for continuous speech recognition the hypothesized word boundaries are often incorrect and exact time aligned substitutions are rare. In a recent work (Tan et al., 2001), word level confidence measures are extracted from N-best sub-hypotheses, namely a method based on a second-best likelihood ratio (SBLR) which has shown to have good properties in rejecting wrong words. In the present work, we have also used this kind of likelihood ratio test from the “word N-best” list that is computed during the Viterbi search when the path is being built. The method was developed in the framework of the (multiple) token-passing model used in the HTK toolkit (Young et al., 2001). We do not consider any language model; however, the recognizer grammar used can be seen as a bigram model.

The paper is organized as follows. In the section 2 we describe briefly the word graph approach and how he implemented it. In section 3 we introduce the proposed confidence measure. The experiments and the results are described in section 4. The last section presents concluding remarks and future work.

2. Word graphs

A word graph is a directed acyclic graph representing recognizer alternatives. A recognition hypothesis for a word w_e ending at a time t is represented as a node $w_e(t)$ in the search space. The edge from a previous node $w_s(t_a)$ to the actual node $w_e(t)$ is represented as $w_{se}(t_a,t)$ and contains all the information for the hypothesized word: the word label and the start and end instants. With this framework, the forward-backward algorithm can be applied in order to obtain the posteriori word probabilities. For the bigram case, all what is needed is to associate to each node the usual alpha and beta probabilities. For the trigram case, although not used in this work, the alpha and beta probabilities can be associated to the edges. This was already noted in (Hacioglu and Ward, 2002). The general case is given in (Wessel et al., 2001).

For the word graph creation we have used the *word N-best* list provided by the HTK software with the usual beam-width pruning. In this case, more than one token is propagated through the network grammar simultaneously. We have used 2-20 tokens. At the end of the decoding, the forward-backward algorithm is applied to the nodes of the produced word graph or lattice. Finally the word or edge probabilities are computed as the final word confidence measure. The scaling of the acoustic model probabilities, as reported in (Wessel et al., 2001), was also used in our work, which improved significantly the confidence measure.

However, posterior word probabilities are a weak confidence measure, because word alternatives with exactly the endpoints t_a and t may be in a small number or may not exist at all in the word graph. A good solution, proposed in (Wessel et al. 2001), accumulates the posterior probabilities of the hypotheses for a word with slightly different starting and end times. We used this confidence measure, namely one referred to as Cmed, in which the posterior probabilities for word w_e that are accumulated, corresponds to the edges $w_{ie}(t_i,t_f)$ that cross the median instant, $(t_a+t)/2$, of the edge under consideration, $w_{se}(t_a,t)$.
We used this confidence measure as a reference for a new proposed method, described in the following section.

3. The confidence measure

During the decoding, there are paths that compete with each other towards the optimal solution. In the word graph, the competition among words can be seen as different starting nodes that reach a common end node of the local best path (Figure 1). At this instant we can give a confidence measure of the best previous word node (starting node), comparing the likelihood of this local best path with the likelihoods of the alternative paths that contain the competing words. If a strong difference exists between the two top likelihoods it may indicate that the last word of the best local path is probably correct. If the two paths have identical likelihoods, there is no evidence which word would be the correct one.

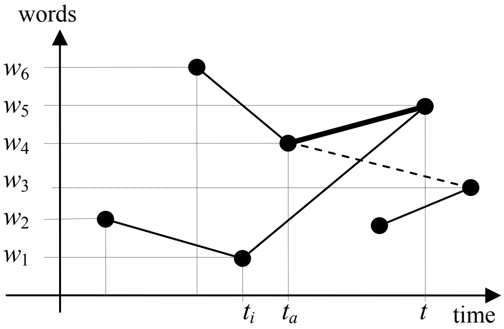

Figure 1: Example of two competing paths that reach a node w$_5$(t). The best edge comes from word w$_4$ (node w$_4$(t_a), bold edge) and the second best edge comes from word w$_1$ (node w$_1$(t_i)). Later on, the node w$_4$(t_a) could be again the stating node of another best edge (dashed line).

Accordingly, we score this starting node with the likelihood ratio between the best and second best hypothesis that reach the end node.

Because may exist different edges coming out from a node, it can be scored several times during the decoding (dashed line in Figure 1). Therefore, we used two ways to define the final node score: one that accumulates the likelihood ratios and takes the logarithm of the result; the other stores only the maximum log likelihood ratio encountered. In the first case the confidence measure is given by

$$C_{SumLR}\left(w_s(t_a)\right)=\log\sum_k r_{sk}(t_k),$$

where $r_{sk}(t_k)$ is the likelihood ratio between the best and second best paths that reach an end node with starting best node $w_s(t_a)$. Actually, this is computed with log likelihood differences which are accumulated iteratively in the log domain. In the second case, the confidence measure is

$$C_{MaxLR}\left(w_s(t_a)\right)=\log\left(\max_k\left(r_{sk}(t_k)\right)\right),$$

which corresponds to the maximum of the log likelihood differences.

One of the main advantages of these confidence measures is that they are performed in only one pass; at the end of the decoding, all the nodes of the word-graph (including those of the Viterbi path) have scores associated with it. A threshold applied to this score can then be used to accept or reject a word in the Viterbi best path.

We tested the proposed confidence measures in a connected digit recognition task, described in the following section.

4. Tests and results

4.1. The database

In the experiments, a Portuguese database was used. This database was recorded locally from female and male speakers, with a normal PC and a unidirectional microphone. It consists in two sets: one training set with a total of 1008 utterances and one testing set with 921 utterances. The training set has 561 utterances having from 1 to 9 digits and 447 with 4 digits. All of the utterances in the testing set have from 1 to 9 digits. All of the training set was segmented and labelled manually.

The speech analysis was accomplished using 26 parameter vectors, from which 13 Mel-frequency Cepstrum Coefficients (including c_0) and 13 Delta coefficients (first-order derivatives), computed from 26 filter bank coefficients, with the frequency ranging from 300 to 3400 Hz. The sampling frequency used was 8 kHz. The frame size was 32 ms and the frame shift was 10 ms. Cepstral Mean Normalization was applied to the cepstral vectors across each input speech file.

4.2. The acoustic models

The HMM set consists of fifteen continuous densities multiple component Gaussian distribution monophones, corresponding to the digits from “zero” to “nine” and 5 garbage models. Each monophone has a left-to-right topology and 8 and 16 mixture components per state. As the digit “one” was shorter, in average, than others, a 6 state HMM was used. The models for “four”, “five”, “seven”, “eight” e “nine” has 9 emitting states and a skip from the 6th emitting state to the last non-emitting state. The models for “zero”, “two”, “three” and “six” has 9 emitting states with no skips. All the filler models have 3 emitting states, having the models for “noise”, “speech” and “silence” a skip from the 3rd to the 1st state.

The same 1008 utterances from the database training corpus were used to train both the keywords and the garbage models.

For the recognition system two task grammars were considered: a simple one with feedback (hereafter referred to as *word loop*), with no string length limit, and another without feedback, permitting a maximum of 9 recognized digits (*constrained grammar*).

The baseline best results are 94% of correct word recognition and 66% of correct word strings. This low performance result is mainly due to the noisy environment conditions of the database acquisition and the relatively low dimension of the database.

4.3. DET curves

The process of estimating a confidence measure can be seen as a statistical hypothesis testing in which a word provided by the recognizer is accepted or rejected. In this process, two errors can occur: false rejection (FR), or type I error (Colton, 1997), if a correct word is rejected; and a false acceptation (FA), or type II error, if a wrong word is accepted. In our case, insertion errors are treated as normal errors, but deletion errors are ignored, because the recognizer output is taken as the alignment reference.

In order to evaluate the performance of the confidence process in a test database, two other measures can be taken into account: the number of words correctly accepted (CA) and correctly rejected (CR). As a metric for the hypothesis test evaluation, the number of false attributions divided by the number of test hypotheses can be used, as in (Mengusoglu and Ris, 2001). However, the trade-off between false acceptations and false rejections, given a threshold, could be used to assess the confidence measure.

The FR rate is defined as the number of correctly recognized words that have been rejected divided by the total number of correctly recognized words. The FA rate is calculated as the number of wrongly accepted words divided by the total number of the recognizer errors.

The Detection Error Tradeoff (DET) curve plots the FR rate versus FA rate for different values of the threshold used for acceptance/rejection.

In (Falavigna et al., 2002) the best threshold is chosen by minimizing the sum of FA rate and FR rate. In our work, we defined the best threshold as the one that gives a minimum distance from the points of the DET curve to the origin (optimum point – OP):

$$OP = \min \sqrt{(\%FA)^2 + (\%FR)^2} \ .$$

4.4 Results

Figure 2 and 3 show the DET curves for our confidence measure, CmaxLR, and the word graph-based confidence measure, Cmed, for acoustics models with 16 mixtures and using 5 tokens in propagation. The figure 2 is for the constraint grammar case and de figure 3 is for the word loop grammar. The marked dots correspond to the OP's defined above. The results of CSumLR (not shown) are almost similar to the CMaxLR measure.

Tables 1 and 2 show the FA and FR ratio on the optimum point for the constrained grammar and word loop grammar respectively. Figure 4 shows CA and CR percentages as a function of the threshold.

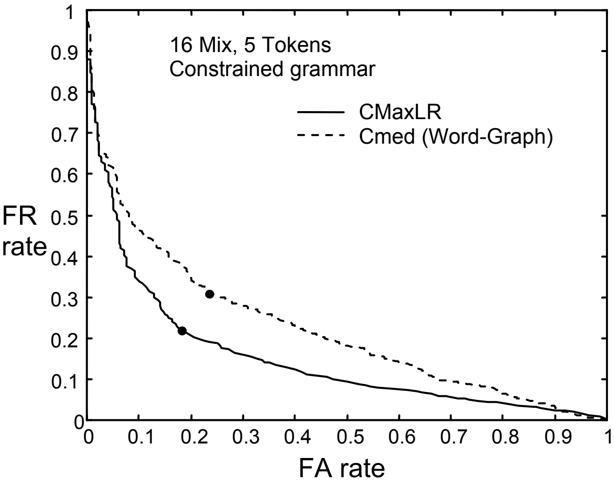

Figure 2: DET curves of Cmed and CMaxLR for 16 component Gaussians, 5 propagated tokens and a constrained grammar.

Constant in all experiments, is that while Cmed presents better results on the word loop grammar, the CMaxLR always reach better results with the constrained grammar.

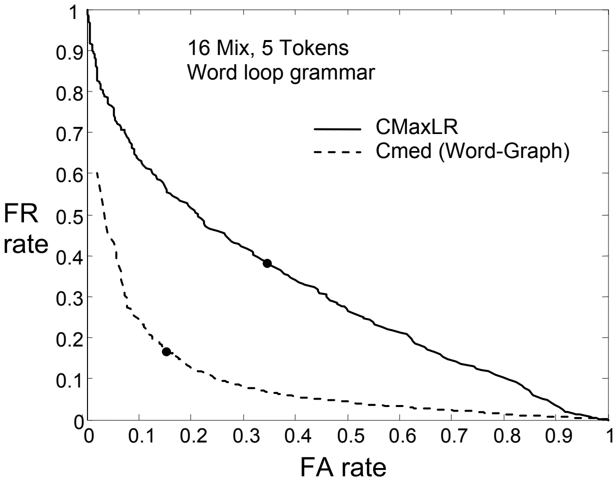

Figure 3: DET curves of Cmed and CMaxLR for 16 component Gaussians, 5 propagated tokens and word loop grammar.

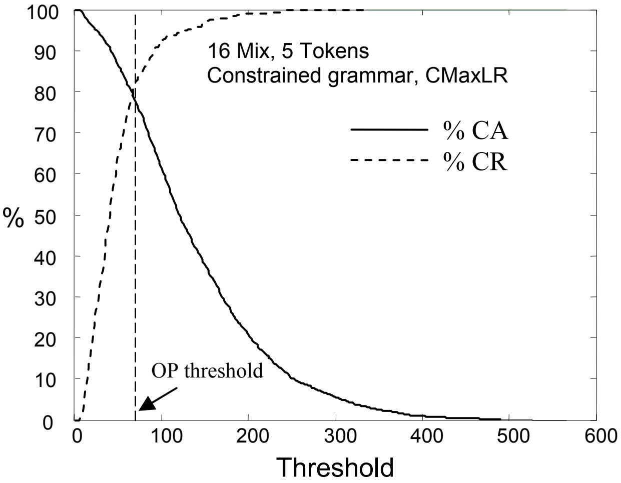

Figure 4: Percentage of Correctly Accepted (CA) and Correctly Rejected (CR) curves as a function of the threshold.

This behavior could be explained in the following way. Unlike the word loop grammar, the constrained grammar limits the number of the edges, because the digit string length is finite. So there are fewer insertions and more deletions errors with the constrained grammar than with the word loop. Even though this increases the correct sentence rate, it reduces the number of correct words. This can explain why word probabilities are lower and Cmed is worse with the constrained grammar. On the other hand, our measure CMaxLR has worse results with the word loop grammar because with this grammar all nodes are potentially competing nodes. This fact degrades significantly de results of the confidence measure.

	% FA	% FR
CMaxLR	17.8	22
Cmed	23	31

Table 1: % FA and % FR witch constrained grammar

	% FA	% FR
CMaxLR	34	38
Cmed	15	17

Table 1: % FA and % FR witch word loop grammar

5. Conclusions and future work

In this paper we have proposed an efficient method to perform word confidence measures in an automatic speech recognition system. The confidence measure is computed during the decoding phase and is based on likelihood ratios between the top hypotheses that reach a word node. Experiments were carried out on a digit database with a connected-digit recognizer. We have compared the confidence measure with the well known and most used word graph probabilities. The results show that with a constrained task grammar the proposed measure is very effective, better than word graph probabilities; however attains a low performance in the case of a word loop grammar. One of the main advantages of the proposed measure is its efficiency, because it is evaluated only in the decoding pass of the recognizer. It could be used with advantage in real time recognizers with low computational resources.

As a future work, we intend to test this confidence measure in other recognition tasks, with larger vocabularies, as well as with other databases for which state-of-art results are available.

References

Colton, L., 1997. *Confidence and Rejection in Automatic Speech Recognition*, PhD thesis, OGI, 1997.

Falavigna, D., R. Gretter, G. Riccardi, 2002. Acoustic and Word Lattice Based Algorithms for Confidence Scores, *Proc. ICSLP* 2002.

Hacioglu, K., W. Ward, 2002. A Concept Graph Based Confidence Measure, *Proc. ICASSP'2002*.

Kemp, T, T. Schaaf, 1997. Estimating Confidence Using Word Lattices, *Proc. Eurospeech'97*, Vol. 2:827-830.

Mengusoglu, E., C. Ris, 2001. Use of Acoustic Prior Information for Confidence Measure in ASR Applications, *Proc. Eurospeech* 2001.

Moreno, P., B. Logan, B. Ray, 2001. A Boosting Approach for Confidence Scoring", *Proc. Eurospeech* 2001.

Ramalingam, C., Y. Gong, L. Netsch, W. Anderson, J. Godfrey, Y. Kao, 1999. Speaker-Dependent Name Dialing in a Car Environment With Out-Of-Vocabulary Rejection, *Proc. ICASSP'99*.

Tan, B., Y. Gu, T. Thomas, 2001. Word Level Confidence Measures Using N-Best Sub-Hypotheses Likelihood Ratio, *Proc. Eurospeech* 2001.

Wessel, F., K. Macherey, R. Schlüter, 1998. Using Word Probabilities as Confidence Measures, *Proc. ICASSP'98*, Vol. 1: 225-228.

Wessel, F., R. Schlüter, K. Macherey, H. Ney, 2001. Confidence Measures for Large Vocabulary Continuous Speech Recognition, *IEEE trans. Speech and Audio Proc.*, 9:288-298.

Willett, D., A. Worm, C. Neukirchen, G. Rigoll, 1998. Confidence Measures for HMM-Based Speech Recognition", *Proc ICSLP'98*.

Young, S. et al., 2001. *The HTK Book*, Cambridge University Press, 2001.