A CONCEPT GRAPH BASED CONFIDENCE MEASURE

Kadri Hacioglu and Wayne Ward

Center for Spoken Language Research
University of Colorado at Boulder
E-mail: {hacioglu,whw}@cslr.colorado.edu

ABSTRACT

In this paper, the confidence measure of a hypothesized word is derived from its posterior probability. In contrast to common approaches, in which N-best lists or word graphs/lattices are used, the posterior probabilities are derived from a concept graph. The concept graph is obtained from a word graph through a partial parsing process using semantic grammars. This approach allows us to use relatively complex and better language models along with acoustic models to compute word posterior probabilities. The language model used is comprised of stochastic context free grammars (one for each concept) and an n-gram concept language model. We show that the posterior probabilities computed on concept graphs outperform those computed on word graphs when used as confidence measures. Results are presented within the context of Colorado University (CU) Communicator System; a telephone-based dialog system for making travel plans by accessing information about flights, hotels and car rentals.

1. INTRODUCTION

In several tasks the output of the speech recognizer is far from perfect. This creates problems in applications that use the transcription directly. For instance, in a dialog system, errors in the output word sequence can lead to a dialog flow that diverges from the user's goal. This results in a longer, or maybe unsuccessful, dialog leaving the user confused, frustrated and dissatisfied with the interaction. So, it is very important for a dialog manager to spot incorrectly recognized words and act accordingly to achive human-like performance. Another example is the unsupervised adaptation of the acoustic models using maximum likelihood linear regression (MLLR). Adaptation with an incorrect transcription degrades the performance. Therefore, one needs a method to spot incorrect words and exclude them from adaptation.

Confidence measures are used to label the words at the output of the speech recognizer as correct or incorrect. The basic approach is to select a set of effective features and find a way of combining them into a confidence measure [1, 2, 3]. Although the combination of several features improves the performance, it is usually not much better than that of the best feature.

The word posterior probabilities have been proposed and used as a confidence measure or as an additional feature [4, 5, 6]. It has been demonstrated that the word posterior probabilities is one of the best features. In those work, those probabilites were calculated using either N-best lists or word graphs/lattices. The knowledge sources were acoustic and n-gram word/class based n-gram language models. Recently, in dialog systems, it has been observed that additional features at higher levels, e.g. parsing or understanding levels, improve the performance significantly [7, 8] . However, those features were not probabilistic. Superiority of posterior probabilities to non-probabilistic features have been clearly illustrated in [9].

In this paper, we incorporate knowledge at understanding level as a relatively complex and sophisticated statistical language model (SLM). The SLM has been developed within a flexible speech understanding framework [10]. It consists of a set of stochastic context free grammars, one for each concept, and dialog context conditioned trigram concept LMs. We use SCFGs to parse the word graph into a concept graph. Then, we compute word posterior probabilities on the concept graph using acoustic models, SCFGs, and trigram concept LMs interpolated with word/class based trigram LM. We claim that this method provides a strong feature that can be used alone or with other features for confidence annotation. We compare a confidence measure computed on concept graphs to that computed on word graphs. We provide results that support our claim.

The paper is organized as follows. In section 2, we summarize the posterior probability computation on word graphs using the forward/backward algorithm. The extension of this method to concept graphs is explained in section 3. The experimental setup and results are presented in section 4. The last section includes concluding remarks and possible future work.

2. WORD PROBABILITIES ON WORD GRAPHS

In this section, we present a forward/backward type algorithm for calculating the word probabilities on word graphs. Let s be the start frame and e be the end frame of a word w in a sequence of words w_1^N that spans a time interval of length T frames. We define a word event as $[w, s, e]$. We are interested in the probability of this event given the acoustic observation o_1^T. This probability should be calculated over all possible word sequences that contain w at the interval [s, e]. However, it is a common practice to restrict the computation to a word graph, as it is a compact representation of the most probable word sequences.

We first explain the word graph generated by our speech recognizer. It is a directed weighted acyclic graph. It is built from the

The work is supported by DARPA through SPAWAR under grant #N66001-00-2-8906.

word lattice created during frame synchronous tree lexicon Viterbi beam search. Its nodes represent unique (w, s) pairs. Edges are labeled with end frames and acoustic model scores, $p(o^e_s/w)$. They point to nodes $(e + 1, w_s)$, where e is the end frame of the word w and w_s is its successor. Note that each unique node can link to more than one nodes, thanks to the possibility of more than one end times and successors of a particular word that starts at frame s. Each path through the word graph is a word sequence that spans the time interval of length T. The set of all paths defines the ensemble on which we calculate the posterior probabilities.

Each word event $[w, s, e]$ corresponds to a set of edges in the word graph. So, the probability of a word event is the total probability of all edges associated with it. In developing the forward/backward type algorithm, we consider the edges as HMM like states . The emission probabilities are the acoustic scores kept at the edges. The transition probabilities are provided by the language model in use. To derive the algorithm we need to define edges uniquely. So, the edge from node (w_i, s) to $(w_{i+1}, e + 1)$ is defined as $E^{(w_{i+1},e+1)}_{(w_i,s)}$. We define the forward probability of an edge as $\alpha(E^{(w_{i+1},e+1)}_{(w_i,s)})$. The following recursion can be used to compute the forward probabilities:

$$\alpha(E^{(w_{i+1},e+1)}_{(w_i,s)}) = P(o^e_s|w_i)\;.\sum_{(w_{i-1},s')} \alpha(E^{(w_i,s)}_{(w_{i-1},s')}) \\ .\;P(w_{i+1}|w_{i-1}, w_i) \qquad (1)$$

Similarly, we define the backward probability of an edge as $\beta(E^{(w_{i+1},e+1)}_{(w_i,s)})$. The recursion for the backward probabilities is

$$\beta(E^{(w_{i+1},e+1)}_{(w_i,s)}) = \sum_{(w_{i+2},e'+1)} \beta(E^{(w_{i+2},e'+1)}_{(w_{i+1},e+1)})\;.\;P(o^{e'}_{e+1}|w_{i+1}) \\ .\;p(w_{i+2}/w_i, w_{i+1}) \qquad (2)$$

Once we have computed the forward-backward probabilities we can calculate the edge posterior probabilities, and in turn, the word posteriors. The posterior probability of an edge can be obtained as

$$p(E^{(w_{i+1},e+1)}_{(w_i,s)}) = \frac{\alpha(E^{(w_{i+1},e+1)}_{(w_i,s)})\;.\;\beta(E^{(w_{i+1},e+1)}_{(w_i,s)})}{p(o^T_1)} \qquad (3)$$

where $p(o^T_1)$ can be obtained as the summation of the forward probabilities of the edges that end at frame T. Equivalently, it can be calculated as the summation of the backward probabilities of the edges that start at the initial frame.

In fact, the probability in (3) is the posterior probability of the word that starts at frame s, ends at frame e and precedes the word w_{i+1}. In [9], it has been demonstrated that the use of this probability as a confidence measure on a hypothesized word does not give satisfactory results. Instead, we use

$$C(w_i) = \sum_{(w_{i+1},e+1)} p(E^{(w_{i+1},e+1)}_{(w_i,s)}) \qquad (4)$$

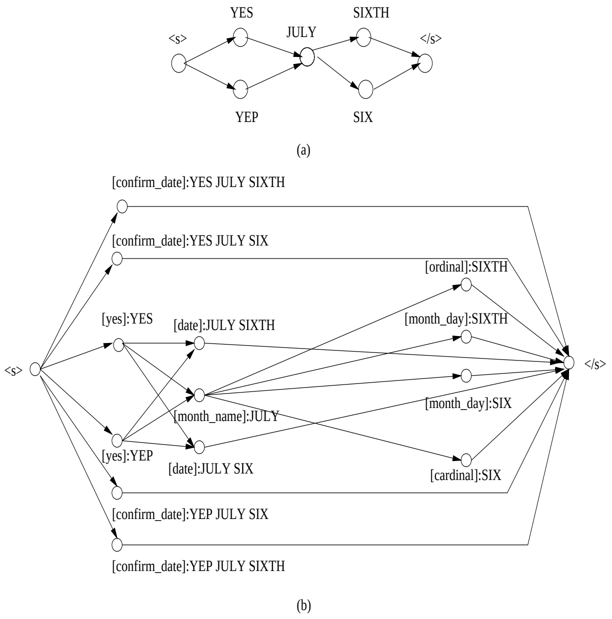

Fig. 1. (a) word graph ,(b) concept graph

This amounts to the summation of the probabilities of the edges outgoing from node (w_i, s). For other possibilities of confidence measures derived from posterior probabilities the reader is referred to [9].

3. CONCEPT/WORD PROBABILITIES ON CONCEPT GRAPHS

The structure of the concept graph is same as the word graph. The nodes are associated with (c, t_s) pairs. The concept c can span a number of words. An example of a concept graph derived from a word graph by partial parsing using a semantic grammar is depicted in Figure 1.

The concepts are classes of phrases with the same meaning. That is, a concept class is a set of phrases that can be used to express that concept (e.g. [date], [yes]). Each concept (except degenerate single word concepts) is written as a context free grammar (CFG) and compiled into a recursive transition network (RTN). The arcs of RTNs are populated with probabilities using a training method based on simple counting and smoothing. So, the multiplication of the arc probabilities that traverse a phrase gives the probability of that phrase. The concept patterns are modeled by n-gram LM conditioned on the dialog context. The dialog context has been taken as the system's last prompt. A detailed discussion of these LMs can be found in [10].

Similar to the word event defined in the preceding section, we define a concept event, $[c, s, e]$, which can also be associated with a set of edges outgoing from node (c, s). Here, the start frame is the start frame of the the first word and the end frame is the end frame of the last word covered by the concept. On each edge, we have an acoustic score , which is the multiplication of the acoustic scores of the words spanned by the concept, and the phrase probability (determined from the concept's SCFG) . One can compute the concept posterior probabilities using the forward backward algorithm described above. On a concept graph, the emmision probability is the acoustic probability multiplied by the phrase probability. The transition probabilities are computed using the trigram concept LM. The corresponding forward backward equations are

$$\alpha(E_{(c_i,s)}^{(c_{i+1},e+1)}) = p(o_s^e|c_i)\ .\ \sum_{(c_{i-1},s')} \alpha(E_{(c_{i-1},s')}^{(c_i,s)}) \\ .\ p(c_{i+1}|c_{i-1},c_i) \qquad (5)$$

$$\beta(E_{(c_i,s)}^{(c_{i+1},e+1)}) = \sum_{(c_{i+2},e'+1)} \beta(E_{(c_{i+1},e+1)}^{(c_{i+2},e'+1)})\ .\ p(o_{e+1}^{e'}|c_{i+1}) \\ .\ p(c_{i+2}/c_i,c_{i+1}) \qquad (6)$$

$$p(E_{(c_i,s)}^{(c_{i+1},e+1}) = \frac{\alpha(E_{(c_i,s)}^{(c_{i+1},e+1)})\ .\ \beta(E_{(c_i,s)}^{(c_{i+1},e+1)})}{p(o_1^T)} \qquad (7)$$

where $p_S(.)$ is the probability conditioned on the dialog context,

$$\begin{aligned} p(o_s^e|c_i) &= p(o_s^e|w_{i,1},\cdots w_{i,L_i})p(w_{i,1},\cdots,w_{i,L_i}|c_i) \\ &= \textstyle\prod_{l=1}^{L_i} p(o_{s_l}^{e_l}|w_{i,l})p(r_{i,l}/c_i) \end{aligned} \qquad (8)$$

, $r_{i,l}$ is the arc of c_i's RTN labeled by the word $w_{i,l}$ and $w_{i,1},\cdots,w_{i,L_i}$ is the L_i-word phrase covered by the concept c_i.

For the sake of simplicity we excluded two things in the derivation of the forward/backward expressions. One is the downscaling of acoustic scores and the other is the interpolation of concept-based SLM with a word/class based SLM. The former is required due to the fact that acoustic scores and LM probabilities have entirely different dynamic ranges. To avoid the domination of summations by a small number of terms, the downscaling of the acoustic scores has been found very useful [9]. Our observations during experiments were on the same line. The interpolation is required to take the advantage of the complementary nature of two different SLMs. We believe that the interpolation partially recovers the loss from context free assumption. The interpolation is performed at the concept/phrase level. That is, the term

$$p(w_{i,1},\cdots,w_{i,L}|c_i)\ .\ p_S(c_i|c_{i-2},c_{i-1})$$

in the expressions above is replaced by

$$(p(w_{i,1},\cdots,w_{i,L_i}|c_i)\ .\ p_S(c_i|c_{i-2},c_{i-1}))^{\lambda} \\ .\ \prod_{l=1}^{L_i} p(w_{i,l}|w_{i,l-2},w_{i,l-1})^{1-\lambda}$$

where $w_{i,0}$ and $w_{i,-1}$ are taken from preceding concepts. That is, $w_{i,0} = w_{i-1,L_{i-1}}$ and $w_{i,-1} = w_{i-1,L_{i-1}-1}$, if $L_{i-1} > 1$. Otherwise, $w_{i,-1} = w_{i-2,L_{i-2}}$. Note that the interpolation is log-linear and λ is the interpolation weight. This interpolation method is selected for two reasons. First, it is easier to implement since the actual implementation deals with the logarithm of probabilities. Second, its performance has been found better than the linear interpolation [11, 12].

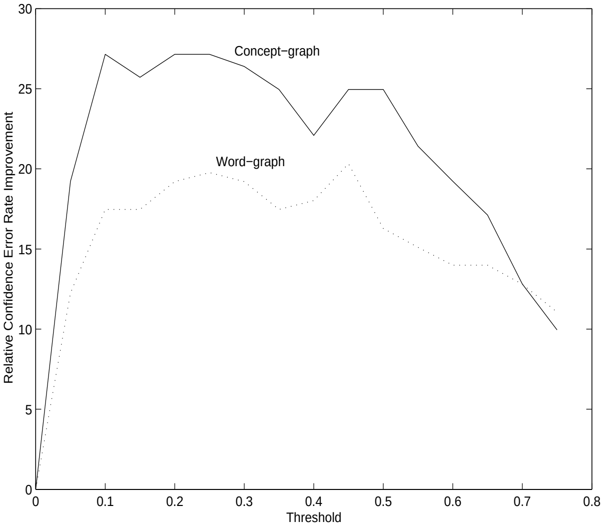

Fig. 2. Relative CER improvement with respect to threshold

Let Cw_i be the set of concepts that include the word event $[w_i,s,e]$. Using the concept posteriors we calculate the confidence of w_i as

$$C(w_i) = \sum_{c_i \in Cw_i} \sum_{(c_i+1,e+1)} p(E_{(c_i,s)}^{(c_{i+1},e+1)}) \qquad (9)$$

4. EXPERIMENTAL RESULTS

We present experimental results on CU Communicator data. The CU communicator system is a dialog system used for flight, hotel and rental car reservations [13]. The data that we experimented with was collected during National Institute of Standards (NIST) 2000 evaluation. It is from a total of 72 calls. The number of female and male callers are 44 and 28, respectively. A total of 1264 sentences were used in the experiments. Of these, 450 sentences were used to optimize scaling, interpolation factors and tagging thresholds. The final results are on the rest of the data.

We used confidence error rates (CERs) and receiver operating characteristics (ROC) curves to evaluate the confidence measures. The CER is defined as

$$CER = \frac{\text{\# incorrectly tagged words}}{\text{\# recognized words}}$$

The baseline CER is obtained assuming that all recognized words are tagged as correct. This is equivalent to the summation of insertions and substitutions divided by the number of recognized words. The ROC curve is the plot of the correct rejection with respect to false rejection. The correct rejection is tagging the incorrect word as incorrect and the false rejection is tagging the correct word as incorrect.

Figure 2 shows relative CER performance improvements over the baseline CER for both word graph and concept graph based word posteriors. It is plotted with respect to the threshold, as the

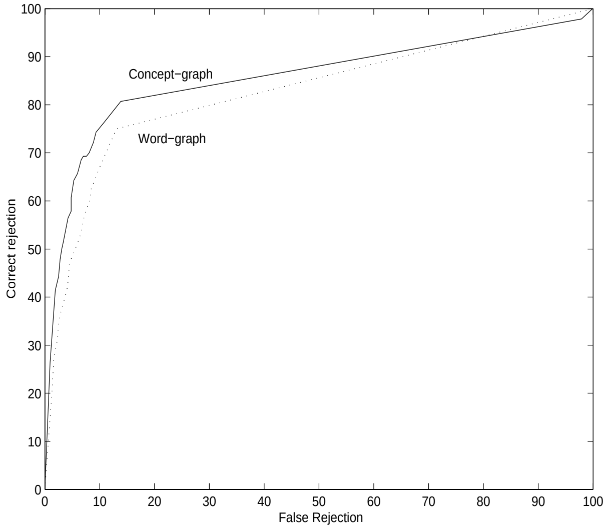

Fig. 3. Receiver operating characteristics (ROC) curve

Table 1. CER and correct rejection (CR) results at 5% false rejection (FR)

Method	Baseline	CER	Reduction	CR
word graph	14.4%	11.8%	18.1%	48.9%
concept graph	11.9%	8.8%	26.1%	63.2%

CER strongly depends on the choice of the tagging threshold. The performances are compared relative to the baseline since two recognizers have different operating points. Figure 3 shows the ROC curve. Both illustrate that the concept graph method performs better.

Table 1, gives baseline CERs, the CERs after confidence annotation, relative reduction in CERs and correct rejection (CR) results at 5% false rejection (FR). All figures clearly show the better performance of the confidence measure computed on concept graphs.

5. CONCLUSIONS

We have presented a confidence measure based on concept/word posterior probabilities computed on concept graphs. We have shown that it outperforms a similar confidence measure based on word graphs. We believe that the improvement is due to the incorporation of higher level knowledge sources (syntactic and semantic constraints) into the computation of posterior probabilities. We plan to use this confidence measure for rescoring the concept graph to improve recognition performance. In addition, the use of this confidence measure in MLLR adaptation is worthy of future research.

6. REFERENCES

[1] S. Cox and R. Rose, "Confidence measures for the switcboard database," in *International Conference of Acoustics, Speech, and Signal Processing*, May 1996, pp. 511–514.

[2] L. Chase, "Word and acoustic confidence annotation for large vocabulary speech recognition," in *Fifth European Conf. on Speech Communication and Technology*, Rhodes, Greece, September 1997, pp. 815–818.

[3] M. Weintraub, F. Beaufays, Z. Rivlin, Y. Konig, and A. Stolcke, "Neural network based measures of confidence for word recognition," in *International Conference of Acoustics, Speech, and Signal Processing*, Munich, Germany, April 1997, pp. 887–990.

[4] T. Kemp and T. Schaaf, "Estimating confidence using word lattices," in *Fifth European Conf. on Speech Communication and Technology*, Rhodes, Greece, September 1997, pp. 827–830.

[5] F. Wessel, K. Macherey, and R.Schluter, "Using word probabilities as confidence measures," in *International Conference of Acoustics, Speech, and Signal Processing*, Seattle, WA, May 1998, pp. 225–228.

[6] F. Wessel, K. Macherey, and H. Ney, "A comparison of word graph and n-best list based confidence measures," in *International Conference of Acoustics, Speech, and Signal Processing*, Seattle, WA, May 1998, pp. 225–228.

[7] R. San-Segundo, B. Pellom, K. Hacioglu, W. Ward, and J.M. Pardo, "Confidence measures for dialogue systems," in *International Conference on Acoustics, Speech, and Signal Processing*, Salt Lake City, Utah, May 2001.

[8] R. Zhang and A. Rudnicky, "Word level confidence annotation using combination of features," in *Seventh European Conf. on Speech Communication and Technology*, Aalborg, Denmark, September 2001, pp. 2105–2108.

[9] F. Wesse, R. Schluter, K. Macherey, and H. Ney, "Confidence measures for large vocabulary continuous speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 3, pp. 288–298, March 2001.

[10] K. Hacioglu and W. Ward, "A word graph interface for a flexible concept based speech understanding framework," in *Seventh European Conf. on Speech Communication and Technology*, Aalborg, Denmark, September 2001, pp. 1775–1778.

[11] D. Klakow, "Log-linear interpolation of language models," in *5-th International Conference on Spoken Language Processing*, Sydney, Australia, 1998, pp. 1695–1699.

[12] K. Hacioglu and W. Ward, "On combining language models: Oracle approach," in *First International Conference on Human Language Technology Research*, San Diego, California,, March 18-21 2001.

[13] W. Ward and B. Pellom, "The CU communicator system," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, Keystone, Colorado, 1999.