CONFIDENCE SCORING FOR SPEECH UNDERSTANDING SYSTEMS[1]

Christine Pao, Philipp Schmid, and James Glass

Spoken Language Systems Group
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139 USA
{pao, phs, jrg}@sls.lcs.mit.edu

ABSTRACT

This research investigates the use of utterance-level features for confidence scoring. Confidence scores are used to accept or reject user utterances in our conversational weather information system [10]. We have developed an automatic labeling algorithm based on a semantic frame comparison between recognized and transcribed orthographies. We explore recognition-based features along with semantic, linguistic, and application-specific features for utterance rejection. Discriminant analysis is used in an iterative process to select the best set of classification features for our utterance rejection sub-system. Experiments show that we can correctly reject over 60% of incorrectly understood utterances while accepting 98% of all correctly understood utterances.

1. INTRODUCTION

Since 1989, our group has been developing conversational systems for human-machine interaction. In the majority of these systems, understanding has been predicated upon either a complete or partial linguistic analysis of one of the top N (e.g., 10) sentences hypothesized by the recognizer. When no such analysis was available, an input utterance would be rejected from further processing. While simple, this method proved effective for both common evaluation and experimental systems [1, 2, 4].

Recently, we have deployed a telephone-based conversational system with much wider access to the general population [10]. Our observation of user behavior with this system led us to believe that a more sophisticated form of rejection was necessary to reduce the number of utterances which were being incorrectly understood and answered. We believed that it would be advantageous for the system to reject a misunderstood, or out-of-domain utterance, rather than provide a possibly lengthy, incorrect response. Therefore, the goal of our research on utterance rejection was to eliminate incorrectly understood sentences as much as possible, while continuing to accept all utterances which were correctly understood.

Different system components can reject a user utterance. The speech recognition component can make use of the likelihood of the acoustic models for a hypothesized word sequence. Phenomena such as out-of-vocabulary or partial words, extraneous noise, or poor signal-to-noise ratio will all tend to result in a poorer match with the acoustic models, and can be a cue to a poor recognition hypothesis. Another cue to a poor hypothesis can be provided by the language model score. Often when confronted by out-of-vocabulary items, the recognizer will hypothesize an unlikely sequence of words in an attempt to match at the acoustic-phonetic level. Finally, when N-best outputs are computed, the relative scores of successive hypotheses can be an indication of recognizer confidence. In addition to the speech recognizer, the natural language component can also provide valuable information. For example, it is extremely useful to know if the utterance can be parsed, and how likely that parse is.

In this study we concentrated on utterance-level features because such features are easily computed and can alleviate the need to combine individual word confidence scores into a meaningful rejection score for the entire utterance.

In this paper we describe our method for automatically tagging training data for rejection or acceptance based on meaning representation. We then present the procedure used to identify sentence level features which could be used for rejection. Finally, we describe a series of classification experiments we have performed using a telephone-based spontaneous-speech corpus.

2. EXPERIMENTAL CORPUS

All experiments were based on telephone data collected from users interacting with our JUPITER weather information system [10]. These data have been continuously collected via a toll-free number since the spring of 1997 using an experimental prototype. To date we have collected and orthographically transcribed over 59,000 utterances from over 10,560 callers. A baseline recognizer was trained on a subset of these data. On an independent test set, the word error rate was 15% [3]. All rejection experiments were based on three sets of data independent from the training data. Final testing was performed on yet another independent test set.

3. AUTOMATIC ANNOTATION

Unlike most work on confidence measures, which is based on word-recognition, we were interested in identifying utterances which were incorrectly *understood*. In our initial work, we manually tagged data, based on examining the top three recognition hypotheses of each utterance and comparing them to the orthographic transcription. Each utterance could be tagged as either ACCEPT, REJECT, or UNSURE based on the similarity between hypothesized and true orthography. The problem with such a procedure was that it was tedious to transcribe a large

[1] This research was supported by DARPA under contract N66001-96-C-8526, monitored through Naval Command, Control and Ocean Surveillance Center.

```
ORTHOGRAPHY:
what what cities do you know in california

{c wh_query
   :topic {q cities
           :quantifier which
           :number "pl"
           :pred {p in
                  :topic {q state
                          :name "california" }
   :domain "Jupiter" }
}
```

```
RECOGNIZER HYPOTHESIS:
what places do you know in california

{c wh_query
   :topic {q cities
           :quantifier which
           :number "pl"
           :pred {p in
                  :topic {q state
                          :name "california" }
   :domain "Jupiter" }
}
```

Figure 1: Examples of matching semantic frames.

```
ORTHOGRAPHY:
yes please how about espaniola

{c what_about
   :random "please"
   :topic {q unknown_city
           :name "espaniola" }
   :subject 1
   :domain "Jupiter" }
}
```

```
RECOGNIZER HYPOTHESIS:
yes please how about aspen you a

{c what_about
   :random "please"
   :topic {q city
           :name "aspen" }
   :subject 1
   :domain "Jupiter" }
}
```

Figure 2: Examples of a mismatch of semantic frames.

set of utterances for training and testing, and the labeling would have to be redone every time a new recognizer was deployed.

We subsequently used these manually annotated data to develop and evaluate an automatic annotation process. This process is based on a comparison of meaning representations produced by a natural language parser [8] rather than on a comparison of reference transcriptions and recognizer hypotheses. This is motivated by our intention to compute a confidence measure based on understanding. The reference transcription and up to N recognition hypotheses for each utterance are parsed, and paired reference/hypothesis semantic frames are generated. If a valid frame is generated for both the reference and the hypothesis, the two frames are compared to determine whether they are identical both in structure and in content [6]. Figure 1 shows an example of two frames that are considered equivalent despite different orthographies. Figure 2 shows an example of a mismatch because of different city names in the orthography and hypothesis. If the reference or the hypothesis, or both, fail to parse, they are not considered to match.

In the case of manual annotation, if one of the top three recognition hypotheses was marked ACCEPT, then the entire list of utterances was accepted; otherwise, the list was rejected. The automatic procedure accepted an utterance if the semantic representation generated from any of the top N recognition hypotheses matched that generated from the orthography.

The automatic annotation agreed with the manual annotation in 1858/2051 cases (90.6% agreement). When the 193 disagreements were examined, 154 were resolved in favor of the automatic annotation and 39 in favor of the manual annotation. Assuming the cases where manual and automatic annotation are in agreement were correctly marked, the automatic annotation is 98.1% accurate. According to the automatic annotation, the training, development, and test sets used for later experiments had correct understanding rates of 63.6%, 56.6%, and 64.9% respectively.

4. FEATURES FOR CONFIDENCE SCORING

We have experimented with 2 types of features for confidence scoring. In addition to the traditional, recognizer-based features such as acoustic scores commonly used in keyword spotting systems [7], we also investigated the use of linguistic and application-specific features (e.g., parse probability) described below.

4.1. Recognition-based Features

Various types of models and parameters are used in todays recognizers. Their likelihood of fit to the data is an indication of confidence and hence can be used for rejection. Commonly used features based on these models are the acoustic and language model scores, the number of words and phones in the hypotheses, and the number of N-best hypotheses. Additionally, recent work in confidence measures suggests that features based on an analysis of the structure of the N-best recognition hypotheses can produce powerful rejection features, such as the A-stabil feature [7], or the posterior log-probabilities of an N-best list [9]. For our experiments, we defined a word score feature which was based on the fraction of N-best sentences in which a word occurred.

4.2. Linguistic and Application-Specific Features

Linguistic features are based on parsing a hypothesis into a syntactic and/or semantic structure, such as a semantic frame. The quality of the parse can be measured by the parse status such as full, partial, or no parse, and the parse probability, when it is available.

4.3. Semantic Features

In our application certain words are semantically more important than others: geography (e.g., city, state, and country names) and weather-related words (e.g., rain, sunshine) are more important than auxiliary verbs for example. Therefore, we have designed

Code	Weight	Word Classes
GEOGRAPHY	3	CITY, CITY_COUNTRY COUNTRY_TYPE PROVINCE, REGION STATE, OCEAN OCEAN_TYPE
CONTENT	2	DAY, DIGIT WEATHER_ATTRIBUTE WEATHER_NOUN, ...
FUNCTION	1	THANKS, QUANT, AUX DO, EXPECTED CURRENT
OTHERS	0	

Table 1: Semantic weights for the different word classes.

"semantic" features that try to measure the amount of information contained in a given utterance (semantic weight) and the difference in semantic content between two sentence hypotheses from the N-best list (semantic distance). Each word in the vocabulary is assigned a weight depending on the word class (taken from the recognizer word-class bigram). Table 1 summarizes the word classes and weights used in our experiments.

The semantic weight of an utterance is the sum of all semantic weights of the words contained within. The semantic distance between two utterances in the N-best list is computed using the Levenshtein algorithm [5]. The insertion, deletions and substitution costs used by the Levenshtein algorithm are dependent on the semantic weight of the words compared; for example, substituting one GEOGRAPHY word for another GEOGRAPHY word contributed to a high semantic distance (the two utterances are referring to different cities and are hence a likely candidate for rejection). We used the semantic weight of the best recognizer hypothesis as well as the semantic distances between the top three hypotheses as possible confidence scoring features.

5. CLASSIFICATION EXPERIMENTS

We used two different approaches for selecting potential feature sets for classification: linear discriminant analysis (LDA), and regression trees as described below. The Fisher LDA was used to create a pool of feature measurements which could be used by a classifier to discriminate between the two classes. The second method grew a regression tree where splits were made to minimize the impurity between the two classes. Classifiers were used at the terminal nodes of confusable cases.

5.1. Fisher Discriminant Analysis

A Fisher LDA classifier was first used to select the best feature set for this classification task. The feature sets were created iteratively. On each iteration, N feature sets from the previous iteration were each augmented with one additional feature from the set of M unused features. The $N * M$ new feature sets were scored using LDA classification on a development set, and the top N feature sets were retained for the next iteration. The LDA threshold for each classifier was set to maintain a false rejection rate of 2% on a development set. The procedure terminated when no additional improvement was found.

A set of 14 features automatically selected by the Fisher criterion are shown in Table 2, in the order in which they were selected. The left column in the table identifies the type of feature used, while the right column indicates which of the N-best outputs were used to compute the feature. For example, an N-best index of 1 indicates that only the first choice hypothesis was used. This feature set achieved 60% correct rejection on the development and test sets. The false rejection rate on the test set increased slightly to 3%. Additional classification experiments using neural network classifiers did not significantly improve the correct rejection rate.

FEATURE	N-best Index
N-gram LM Score	1
Full/Robust/No Parse	1
Total number of hypotheses	all
Average acoustic word score	1
# of hypotheses with no parse before the first full parse	all
Difference of word scores (hyp 1 - hyp 2)	1 & 2
Ngram LM score/# of words	1
# of hypotheses with no parse	all
Total acoustic score	1
Sum of word scores ≥ 0.5	all
Sum of word scores ≥ 0.5	1
Acoustic score (hyp 1 - hyp 2)	1 & 2
Acoustic score/# of phones	1
# words with word score ≥ 0.5	all

Table 2: Feature set selected via Fisher discriminant analysis.

5.2. Decision Tree Analysis

One problem with the Fisher classifier is that the different features are combined into a single measure, making it more difficult to understand the importance of, and interaction between the individual features. We decided it would be interesting to see how individual features could be used to partition the feature space. The basic idea was to split off sets of tokens, where the split set had a high percentage of accept or reject tokens (i.e., high purity). This might allow us to provide more useful feedback information as to why a particular utterance was rejected.

Regression trees were created by searching for the feature vector which could split a node (i.e., data subset) to meet purity and size requirements. After a node had been split, all remaining nodes which did not meet the purity requirement were merged together for subsequent splitting. If there was no split which met both the purity and minimum size requirements, the purity constraints were relaxed and the search was repeated. A development set was used for cross-validation purposes.

While the resulting regression tree structure could be used directly as a classifier for utterance rejection, there was a greater degradation in performance when moving from development to test sets, when compared to our initial Fisher classifier experiments. However, we did find the regression trees helpful for identifying features which were useful for confidence scoring.

System	Manual ACCEPT		Manual REJECT		Total
ACCEPT	14,075	97.2%	3,980	36.7%	18,055
REJECT	412	2.8%	6,879	63.3%	7,291
Total	14,487		10,859		25,346

Table 3: Confidence scoring results: the correct decision was made in 82.7% of the cases ((14,075 + 6,879) / 25,346).

Orthography	Best Hypothesis
south texas	how is texas
what is the shrimp catch like in new orleans louisiana	what is that i like in new orleans lousiana
how about sydney	how about today

Table 4: Examples of incorrectly accepted utterances.

6. DISCUSSION

During a 4 month period in the spring of 1998 our group collected over 25,000 utterances in the JUPITER domain. We determined that our rejection component incorrectly rejected 2.8% and correctly rejected 63.3% of all utterances, for a total of 82.7% correct ACCEPT/REJECT decision. Table 3 summarizes the classification results using a Fisher LDA with the best feature set. Note that the overall understanding rate of both within-domain and out-of-domain queries was 57.2%, based on the automatic understanding of the orthographic transcriptions. The overall understanding rates for both our training and test data are considerably lower than the near 80% understanding we report on within-domain data subsets [6]. This phenomenon reflects the presence of spontaneous speech events, noise, and out-of-vocabulary words in the out-of-domain queries.

Table 4 lists examples of incorrectly accepted utterances, which account for about 40% of all accepted utterances. In many cases there is an unknown or misrecognized city name contained in the recognizer hypothesis. In other cases the system accepted utterances containing non-speech events or out-of-domain requests such as the second example in Table 4.

Further analysis of the system's behavior revealed that if an utterance is correctly accepted, the following utterance is most likely to be accepted and understood as well. This is presumably because system performance is better if the user knows how to talk to the system, and the system likewise provides positive reinforcement to the user by answering correctly. Similarly, if an utterance is correctly rejected, the following utterance is most likely to be rejected. This might be an indication that our current system response to a rejected utterances ("Sorry, I'm not sure what you said") is not very helpful; the system is most likely to reject the following utterance after the first rejection message. However, the system is much more likely to accept an utterance after the second rejection message, which is more helpful. The worst understanding rates are after the first rejection message and after an utterance is incorrectly accepted after multiple failures. In general, system performance is the worst after multiple failures – if the system is having trouble understanding the user, it continues to have trouble.

It is interesting to note that users are more likely to hang up after the first rejection message than after they have received multiple rejection messages, particularly if the utterance was rejected incorrectly, perhaps a sign that persistence pays off!

7. FUTURE WORK

The analysis of the user's behavior to rejected utterances suggests that more informative feedback is needed in order to prevent error spirals. Therefore we intend to add word level confidence measures to detect early problems with certain content words and hence will be able to say to the users "Did you say Boston, Massachusetts or Austin, Texas". Similarly, we hope to be able to use the decision tree described earlier as another source of information for improved user feedback.

8. REFERENCES

1. J. Glass, G. Flammia, D. Goodine, M. Phillips, J. Polifroni, S. Sakai, S. Seneff, and V. Zue, "Multilingual spoken language understanding in the MIT Voyager system," in *Speech Communication*, pp. 1-18, vol. 17, 1995.

2. J. Glass, D. Goddeau, L. Hetherington, M. McCandless, C. Pao, M. Phillips, J. Polifroni, S. Seneff, and V. Zue, "The MIT ATIS System: December 1994 Progress Report," in *Proc. Spoken Language Systems Technology Workshop*, pp. 252–256, Austin, Texas, 1995.

3. J. Glass and T. Hazen, "Telephone-Based Conversational Speech Recognition in the JUPITER Domain," *in these Proceedings*, Sydney, Australia, 1998.

4. D. Goddeau, E. Brill, J. Glass, C. Pao, M. Phillips, J. Polifroni, S. Seneff, and V. Zue, "GALAXY: A Human-Language Interface to On-Line Travel Information," in *Proc. ICSLP*, pp. 707–710, Yokohama, 1994.

5. V. Levenshtein, "Binary Codes Capable of Correcting Deletions, Insertions and Reversals," in *Soviet Physics Doklady*, vol. 10, no. 8, pp. 707–710, 1966.

6. J. Polifroni, S. Seneff, J. Glass, and T. Hazen, "Evaluation Methodology for a Telephone-Based Conversational System," in *Proc. 1st International Conference on Language Resources and Evaluation*, pp. 43–49, Granada, Spain, 1998.

7. T. Schaaf and T. Kemp, "Confidence Measures for Spontaneous Speech Recognition," in *Proc. ICASSP*, pp. 875–878, Atlanta, GA, 1996.

8. S. Seneff, "TINA: A Natural Language System for Spoken Language Applications," in *Computational Linguistics*, vol. 18, no. 1, pp. 61–86, 1992.

9. M. Weintraub, F. Beaufays, Z. Rivlin, Y. Konig, and A. Stolcke, "Neural–Network based Measures of Confidence for Word Recognition," in *Proc. ICASSP*, pp. 887–890, Munich, Germany, 1997.

10. V. Zue, S. Seneff, J. Glass, L. Hetherington, E. Hurley, H. Meng, C. Pao, J. Polifroni, R. Schloming, and P. Schmid, "From Interface to Content: Translingual Access and Delivery of On-line Information," in *Proc. Eurospeech*, pp. 2227–2230, Rhodes, Greece, 1997.