INTEGRATING RECOGNITION CONFIDENCE SCORING WITH LANGUAGE UNDERSTANDING AND DIALOGUE MODELING[1]

Timothy J. Hazen, Theresa Burianek, Joseph Polifroni and Stephanie Seneff

Spoken Language Systems Group
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139, USA

ABSTRACT

In this paper we present a method for integrating confidence scores into the understanding and dialogue components of a speech understanding system. The understanding component of our system receives an n-best list of recognition hypotheses augmented with word-level confidence scores. The confidence scores are used by the understanding component to hypothesize when words in a recognizer's n-best list have been misrecognized. The understanding component has the ability to predict the semantic class of misrecognized words based on the surrounding context and also to suggest when key words which may have been misunderstood should be re-confirmed by the user. The output of the understanding component is passed onto a dialogue control component which can act on various suggestions made by the understanding component. To evaluate the system, experiments were conducted using the JUPITER weather information system. Evaluation was performed at the understanding level using key-value pair concept error rate as the evaluation metric. When word confidence scores were integrated into the understanding component, the concept error rate was reduced by 35%.

1. INTRODUCTION

The primary goal of the Spoken Language Systems Group is to conduct research leading to the development of conversational systems for human-machine interaction. These systems, such as the JUPITER weather information server [6], must not only recognize the words which are spoken but also understand the user's query and respond accordingly. Unfortunately, the presence of incorrectly recognized words may cause the system to misunderstand a user's request, possibly resulting in the execution of an undesirable action. To help alleviate the problems associated with misrecognized words, a system should consider the *confidence* the recognizer has in its word string hypotheses. It is important for a conversational system to be able to determine when a misrecognition could harm the understanding of a user's input utterance and to take an appropriate action when there is a reasonably high likelihood that the system has misunderstood the user's request.

Over the past several years we have been investigating the problem of confidence scoring within our recognizer [2]. Our confidence scoring technique examines a set of features extracted from the recognizer's computation of hypothesized words and sentences. These features are then passed into a confidence classifier which generates a confidence likelihood score that can be used to make an accept/reject decision for each hypothesized word. Using this approach our recognizer is now capable of producing a reasonably accurate estimate of the likelihood that a hypothesized word is correct.

With the availability of recognizer confidence scores, our attention has shifted towards developing methods to integrate these scores into the understanding and dialogue management components of the system. It is our goal to be able to utilize the confidences scores to determine when errors that may lead to a misunderstanding have occurred and to take appropriate actions to recover from these errors. To provide an example, suppose a user asks JUPITER the following question:

> *what is the forecast for paramus park new jersey*

As it happens, the JUPITER speech recognizer does not have the word *paramus* in its vocabulary. As such, the recognizer will provide its best guess using the words it knows. Thus, it might hypothesize the following query:

> *what is the forecast for* **paris** *park new jersey*

Using confidence scoring techniques the JUPITER recognizer should determine that the word **paris** was not a reliable hypothesis. It could then mark this word as a potentially misrecognized word when passing the utterance on to the understanding component of the system. At that point the understanding component would need to be able to determine that the user is looking for the forecast for some place in New Jersey, but that the name of the place was misrecognized. Using this information the system could then prompt the user with the list of places in New Jersey for which it knows forecasts. The system might also prompt the user to spell the name of the city and learn it for future use.

To create a system capable of the actions described above, we have developed several methods for incorporating the confidence scores generated by our recognizer into the language understanding and dialogue modeling components of our system. Our goal has been to enable these components to make informed decisions about the actions that should be taken when the confidence scores indicate potential errors in the recognizer's hypotheses. In this paper we present the details of our approach to this problem and present experimental results demonstrating the capabilities of our techniques.

[1] This research was supported by DARPA under contract N66001-99-1-8904, monitored through Naval Command, Control and Ocean Surveillance Center.

Standard n-best list with word confidence scores:

what_is	6.13	the	5.48	forecast	6.88	for	5.43	**paris**	**-0.03**	park	4.41	new_jersey	4.35
what_is	6.13	the	5.48	forecast	6.88	for	4.47	**hyannis**	**-0.16**	park	4.41	new_jersey	4.35
what_is	6.13	the	5.48	forecast	6.88	for	5.12	**venice**	**-1.49**	park	4.41	new_jersey	4.35

n-best list with hard rejection:

what_is	6.13	the	5.48	forecast	6.88	for	5.43	***reject***	**0.00**	park	4.41	new_jersey	4.35
what_is	6.13	the	5.48	forecast	6.88	for	4.47	***reject***	**0.00**	park	4.41	new_jersey	4.35
what_is	6.13	the	5.48	forecast	6.88	for	5.12	***reject***	**0.00**	park	4.41	new_jersey	4.35

n-best list with optional rejection:

what_is	6.13	the	5.48	forecast	6.88	for	5.43	***reject***	**0.00**	park	4.41	new_jersey	4.35
what_is	6.13	the	5.48	forecast	6.88	for	5.43	**paris**	**-0.03**	park	4.41	new_jersey	4.35
what_is	6.13	the	5.48	forecast	6.88	for	4.47	***reject***	**0.00**	park	4.41	new_jersey	4.35
what_is	6.13	the	5.48	forecast	6.88	for	4.47	**hyannis**	**-0.16**	park	4.41	new_jersey	4.35
what_is	6.13	the	5.48	forecast	6.88	for	5.12	***reject***	**0.00**	park	4.41	new_jersey	4.35
what_is	6.13	the	5.48	forecast	6.88	for	5.12	**venice**	**-1.49**	park	4.41	new_jersey	4.35

Figure 1: Example n-best lists with word confidence scores for the utterance *"What is the forecast for Paramus Park, New Jersey?"*.

2. CONFIDENCE SCORE INTEGRATION

2.1. Overview

The goal of this research was to develop methods for integrating recognition confidence scores into the language understanding and dialogue modeling components of a conversational speech system. This research was conducted using the JUPITER weather information conversational speech system [6]. JUPITER is a mixed-initiative system which allows users to enquire about weather forecast information for over 500 cities around the world. The system utilizes the GALAXY-II spoken language system architecture to control the flow of information between the various speech components employed by the system [5]. These components include the SUMMIT speech recognizer [1] and the TINA natural language understanding system [4].

The SUMMIT speech recognizer has the capability to output probabilistic *word*-level confidence scores for each word in each sentence hypothesis in the n-best list generated by the recognizer. In past work, we have shown that SUMMIT also has the capability to generate *utterance*-level confidence scores which reflect the performance of the recognizer over the entire utterance. However, recent experiments have shown that the utterance level scores are redundant when used in conjunction with word level scores, and hence do not improve system performance. Thus, the research in this paper only uses word-level confidence scores. The word-level scores are zero-based such that hypotheses with positive scores are *accepted* while hypotheses with negative scores are candidates for *rejection*.

The TINA natural language understanding system utilizes a semantically-tagged probabilistic context free grammar to parse and understand a user's query. TINA takes the n-best list with word scores and collapses it down to a word graph. TINA then searches through the word graph attempting to parse potential sentences. TINA selects the most appropriate interpretation of the user's query (based on a combination of the word confidence scores generated from the recognition process and the parse scores computed by TINA). From the selected sentence, TINA generates a semantic representation in the form of a key-value concept pair which captures the information relevant for retrieving the correct answer to the user's query from a database.

2.2. Augmenting Recognition Output

In order for the TINA understanding component to utilize the confidence scores generated by SUMMIT, methods for interpreting these scores and integrating them into TINA had to be developed. As mentioned above, TINA accepts an n-best list augmented with confidence scores as its input. The top section of Figure 1 shows an example n-best list.

After a standard n-best list with confidence scores is generated, additional modifications can be made to the n-best list as a preprocessing stage before understanding is performed. The n-best list can be augmented with markings indicating when words are highly likely to have been misrecognized (and hence should be rejected) or moderately likely to have been misrecognized (and hence should be confirmed before any action is taken). Support for these markings must then be incorporated into the language understanding and dialogue components of the system.

The first type of modification that is performed during the preprocessing is to mark words with poor confidence scores for rejection. There are two methods of word rejection that can be employed. The first method is *hard rejection*, in which all words which fall below a particular level of confidence are rejected outright and replaced in the n-best list with a *rejected word marker*. The second section of Figure 1 shows the example n-best with all words with negative confidence scores replaced with the marker ***reject*** and their scores set to a neutral score of zero.

Because the process of *hard rejection* is irreversible (i.e., a hypothesized word cannot be recovered once it is rejected), an alternative approach called *optional rejection* can be used. The third section of Figure 1 shows an example n-best list using op-

Standard n-best list with word confidence scores:

what_is	6.13	the	5.48	forecast	6.88	for	5.61	**boston**	**0.95**
what_is	6.13	the	5.48	forecast	6.88	for	4.23	**austin**	**0.21**
what_is	6.13	the	5.48	forecast	6.88	for	5.09	**charleston**	**-1.52**

n-best list using both hard rejection and confirmation:

what_is	6.13	the	5.48	forecast	6.88	for	5.61	***confirm* boston**	**0.95**
what_is	6.13	the	5.48	forecast	6.88	for	4.23	***confirm* austin**	**0.21**
what_is	6.13	the	5.48	forecast	6.88	for	5.09	***reject***	**0.00**

Figure 2: Example n-best lists with word confidence scores for the utterance *"What is the forecast for Boston?"*.

tional rejection. This list is essentially the combination of the first two lists. Using optional rejection, poorly scoring words are retained in the final n-best list but must compete with the rejected word markers they generate, which have a higher score.

In cases where an important hypothesized content word has a low enough confidence score so as to call its correctness into question, but not so poor a score that it should be rejected outright, it may be desirable to accept the word but require a follow-up confirmation from the user to verify that the word is indeed correct. To accommodate this type of action, important key words can be marked for confirmation in the n-best list if their scores fall below a certain threshold but are not outright rejected. Figure 2 demonstrates how the important content words Boston and Austin both receive small positive confidence scores and are tagged with the ***confirm*** marker while the word Charleston is rejected since its score is negative. Only city names are marked for confirmation in the experiments in this paper.

2.3. Augmenting the Understanding Grammar

In order to utilize the ***reject*** and ***confirm*** markers, the understanding grammar must be augmented to accommodate these markers. For our experiments with JUPITER, only three modifications to the grammar were made. First, to handle the ***reject*** marker the grammar was modified to accept this marker in lieu of a word in specific contexts. Specifically, the grammar was adjusted to allow rejected words to be parsed as *unknown city names* in contexts where the rejected word was almost certainly a city name. In Figure 1, for example, the word sequence ***reject* park** would be parsed as an unknown city name.

The second adjustment to the grammar was to allow rejected words appearing anywhere in the sentence to be skipped when the parser is attempting to find a robust or partial parse. This allows the parser to concentrate on only the portions of the utterance which were recognized with high confidence. This modification is especially useful for eliminating problems that result from spurious sounds or speech at the beginning and/or end of an utterance.

Finally, to handle the ***confirm*** marker, the grammar was modified to accept this marker at the start of any city name. The marker thus becomes part of a city's name when it is passed to the dialogue manager. The dialogue manager can then check for this marker when any city name is passed to it, and ask an appropriate confirmation question when it is present.

3. EXPERIMENTAL RESULTS

3.1. Understanding Using Word Rejection

The confidence scoring methods described above have been incorporated into the JUPITER system and replace a pre-existing heuristic word scoring method utilized by TINA. To examine the effects of word confidence scoring on language understanding, the JUPITER system was evaluated on 2388 test utterances under four different conditions: (1) using our original system which did not utilize word confidence scores, (2) using the new system which utilizes word confidence scores but does not perform any rejection, (3) using the new system with optional word rejection, and (4) using the new system with hard word rejection. The confirmation markers are ignored in this evaluation. The system is evaluated using key-value pair concept error rate [3]. The results are shown in Table 1 in terms of substitution, insertion, deletion, and total error rates. For these experiments, a substitution error occurs when a test utterance has a key-value pair where the key matches a key-value pair in the correct answer, but the value in the pair is different. An insertion occurs when a key-value concept is erroneously inserted. Likewise, a deletion occurs when a key-value concept is erroneously deleted.

Examining Table 1 yields several important observations. First, the new system using the probabilistic word confidence scores has an error rate which is an 8% reduction (from 28.5% to 26.2%) from the error rate of the original system using the heuristic word scores. However, both systems suffer from excessive insertion errors when no rejection is utilized. This is primarily the result of the understanding component's aggressive effort to find a reasonable interpretation of an utterance from any of the hypotheses in the n-best list. Without rejection, the understanding component can latch onto any hypothesis which produces a parse regardless of whether or not the recognizer is confident in the hypothesis. This generally produces the correct answer when the user is cooperative, speaks clearly and stays within domain. However, this approach yields many insertions when the utterance is out of domain, has unknown words, or has artifacts which cause difficulty for the recognizer.

Next, the use of either optional or hard word rejection produces a significant improvement in the total error rate. While the total error rates for optional word rejection versus hard word rejection are virtually the same, the nature of the underlying errors is slightly different. Using optional word rejection, the insertion error rate remains higher than the deletion error rate. However,

Experimental Conditions	Error Rates (%) Sub.	 Ins.	 Del.	 Total
Original system	2.2	19.9	6.3	28.5
New system w/o reject.	2.1	18.1	6.1	26.2
+ optional word reject.	1.3	8.9	8.5	18.7
+ hard word rejection	1.0	7.0	10.6	18.6

Table 1: Understanding error rates as confidence scores and different levels of confidence rejection are added to the system.

hard word rejection produces a result where deletions outnumber insertions. The relative desirability of each method would thus be dependent on whether insertion errors are more harmful to the user's interaction with the system than deletions. The addition of word rejection reduces the error rate by 29% (from 26.2% to 18.6%) from the system that doesn't use rejection. Overall, the use of word confidence scores and rejection within the understanding component of the system reduces the understanding error rate by 35% (from 28.5% to 18.6%).

3.2. Understanding Using Confirmation

Evaluating the effectiveness of the confirmation markers cannot be done based only on an evaluation of understanding results. The true worth of the confirmation markers is best determined from user studies of a system which incorporates confirmation dialogue actions. The question of their effectiveness is closely tied to the user's feelings about the usefulness/annoyance levels of confirming various pieces of information before proceeding with an action. However, basic statistics for various confirmation threshold levels can be quoted to provide an idea about how the confirmation markers perform during understanding.

The frequency of confirmations in a dialogue can be tuned based on a confidence score threshold placed on the spoken words that the system may wish to confirm. Table 2 shows statistics collected on city name key-value pairs hypothesized by a system using a moderate confirmation threshold and also employing optional word rejection. Thus, there are three possible actions that can be taken when a city key-value pair is hypothesized: (1) the system accepts the city name and proceeds on, (2) the system requires a confirmation of the city name from the user, and (3) the system outright rejects the city name. The statistics show that city names are accepted 74% of the time with a false acceptance rate of only 3.3%. Rejections of city names occur only 4.8% of the time with a false rejection rate of only 1.9%. These low error rates come at the expense of requiring an additional confirmation from the user 21% of the time, of which 9.8% out of the 21% of confirmations catch city name errors before an incorrect action can occur (assuming the system receives proper confirmation from the user that the city name has indeed been misrecognized). The low false alarm and false acceptance error rates can be reduced further if the confirmation confidence thresholds are expanded to allow even more confirmation requests.

4. DIALOGUE MODELING ISSUES

At this time, we are just beginning to consider the dialogue modeling issues involved in utilizing the confidence scoring techniques that we have presented here. We have only recently begun adding dialogue actions to our system to handle the use of rejection and confirmation as proposed by the understanding component. For example, when city names are rejected by the system but the user has provided a state or country name, the system assists the user by providing a list of cities that the system knows about in that particular state or country. While it is clear that the rejection and confirmation techniques improve the understanding error rates of the system, the real test will be to utilize the confidence scoring information effectively during live conversations with users. This work is on-going.

Hyp. City Correct	Dialogue Action	# of Hyps.	% of Hyps.
Yes	Accept	733	70.5%
Yes	Confirm	120	11.5%
Yes	Reject	20	1.9%
No	Accept	34	3.3%
No	Confirm	102	9.8%
No	Reject	30	2.9%

Table 2: Confirmation statistics compiled for city name key value pairs hypothesized by the system,

5. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a method for integrating recognition confidence scores into the language understanding component of the system. By integrating the word confidence scores into the understanding component of the JUPITER weather information system, we were able to reduce the understanding error rate by 35% using only word rejection techniques. We have also added into the understanding component of the system the ability to request confirmations based on confidence scores. By allowing the system to request confirmation on hypothesized city name concept key-value pairs, the false rejection/false acceptance error rate can be reduced to 5% at the expense of an additional confirmation query from the computer for 21% of the hypothesized city names.

As part of our on-going research efforts we hope to fully integrate the confidence scoring techniques presented in this paper with the dialogue manager of our system in the coming months. This will allow us to utilize these techniques in our publicly available systems and conduct user studies to determine the optimal set of dialogue actions to take under various circumstances.

6. REFERENCES

1. J. Glass, J. Chang, and M. McCandless, "A probabilistic framework for feature-based speech recognition," In *Proc. of ICSLP*, Philadelphia, 1996.

2. T. Hazen, *et al*, "Recognition confidence scoring for use in speech understanding systems," In *Proc. of ISCA ASR2000 Tutorial and Research Workshop*, Paris, 2000.

3. J. Polifroni, *et al*, "Evaluation methodology for a telephone-based conversational system," In *Proc. Int. Conf. on Language Resources and Evaluation*, Granada, Spain, 1998.

4. S. Seneff, "TINA: A natural language system for spoken language applications," *Computational Linguistics*, vol. 18, no. 1, March 1992.

5. S. Seneff, *et al*, "GALAXY-II: A reference architecture for conversational system development," In *Proc. of ICSLP*, Sydney, 1998.

6. V. Zue, *et al*, "JUPITER: A telephone-based conversational interface for weather information," *IEEE Trans. on Speech and Audio Processing*, vol. 8, no. 1, January 2000.