INCORPORATING DISCOURSE FEATURES INTO CONFIDENCE SCORING OF INTENTION RECOGNITION RESULTS IN SPOKEN DIALOGUE SYSTEMS

Ryuichiro Higashinaka, Katsuhito Sudoh, and Mikio Nakano[*]

NTT Communication Science Laboratories, Nippon Telegraph and Telephone Corporation
2-4 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237, Japan
{rh,sudoh,nakano}@atom.brl.ntt.co.jp

ABSTRACT

This paper proposes a method for the confidence scoring of intention recognition results in spoken dialogue systems. To achieve tasks, a spoken dialogue system has to recognize user intentions. However, because of speech recognition errors and ambiguity in user utterances, it sometimes has difficulty recognizing them correctly. Confidence scoring allows errors to be detected in intention recognition results and has proved useful for dialogue management. Conventional methods use the features obtained from speech recognition results for single utterances for confidence scoring. However, this may be insufficient since the intention recognition result is a result of discourse processing. We propose incorporating discourse features for a more accurate confidence scoring of intention recognition results. Experimental results show that incorporating discourse features significantly improves the confidence scoring.

1. INTRODUCTION

For a spoken dialogue system to achieve certain tasks while conversing with users, the system has to recognize user intentions correctly. Here, we use the term *user intention* to express the information that the user has to convey to the system in order to achieve his/her goal, such as extracting some particular information from the system. Since users do not always convey their intentions in one utterance and speech recognition errors might occur, the system and the user normally have to exchange several utterances before the system recognizes the user's true intention. This paper addresses this interactive intention recognition process, focusing on the types of tasks in which intention recognition results are represented by *frames* that consist of *slot-value pairs* [1]. We also assume that the slots are filled with words in speech recognition hypotheses as in many speech applications.

In such interactive intention recognition, the system updates the intention recognition result after each user utterance. Fig. 1 shows how the intention recognition result is updated in the course of a dialogue in a weather information system. In the example, "tomorrow" was misrecognized as "today" by the speech recognizer (U1), causing the system to have an incorrect value for *date* (F2). The misunderstood item was later corrected by the user (U3), who noticed the error in the intention recognition result because of the system's incorrect confirmation request (S3). Through the interactive process with the user, the intention recognition results get closer to the correct user intention (F1-F4).

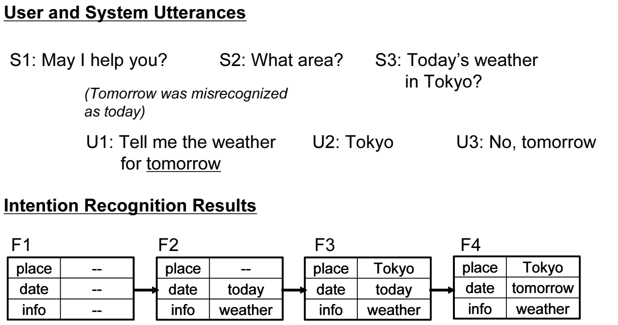

Fig. 1. Updating an intention recognition result. (S, U and F indicate a system utterance, a user utterance, and a frame, respectively.)

Based on the intention recognition result, the system performs dialogue management; namely, it decides what utterances it should produce. Good dialogue management guides a dialogue smoothly, accelerating task completion. To improve task completion, one simple approach for dialogue management is to confirm every item in the slots until all items in them are acknowledged by the user. However, too many confirmations are likely to make dialogues tedious. On the other hand, when the system does not confirm at all, the system is likely to deliver undesired information based on incorrectly recognized items. The system needs to find a balance between too many and too few confirmations.

Recently, *confidence scoring* has been applied to detect errors in intention recognition results and has proved useful for dialogue management [2, 3, 4]. Confidence scoring enables the system to avoid unnecessary confirmations and ask questions on unfilled slots preferentially. Current confidence scoring for slots uses the confidence of words that fill the slots. For example, to obtain the confidence of the slot for the date of F4 in Fig. 1, the word confidence of "tomorrow" in U3 is used. The word confidence is the acoustic and linguistic reliability of the word and typically calculated using various features of speech recognition results. However, since the intention recognition result is the outcome of exchanges of utterances between the user and system, namely, a discourse, using only the speech recognition results of single utterances may not be sufficient.

This paper proposes incorporating discourse features into confidence scoring of intention recognition results. We use both the acoustic and language model features of words that fill the slots and the discourse information concerning the slots to achieve more accurate confidence scoring.

[*] Currently with Honda Research Institute Japan, 8-1, Honcho, Wako-shi, Saitama 351-0114, Japan. nakano@jp.honda-ri.com

2. CONVENTIONAL METHODS

Conventional methods use the confidence of a word in a speech recognition result for the confidence of a slot. We explain how the confidence of a word is obtained from the speech recognition result. Two approaches have typically been used.

One uses a score that the speech recognizer outputs for words, such as the total acoustic and language model score or the word posterior probability [5]. The other uses a confidence score that a *confidence model* outputs [6, 7, 8]. A confidence model is a kind of a classifier that scores elements in speech recognition results based on training data. In the case of a word, each word in the speech recognition hypotheses is labeled correct/incorrect and various features, such as acoustic and language model features concerning the word, are extracted. Then, a confidence model is trained in such a way that the label can be accurately predicted from the features. Even though using the scores that the speech recognizer outputs requires no training, the confidence model approach, which allows the combination of multiple features, tends to be frequently used in research and development, for which accurate confidence scoring is necessary. Although Pradhan et al. use system prompt types before user utterances as one of the features for confidence model training [7] and their approach can be seen as incorporating discourse information, they only focus on speech recognition results, not discourse understanding results, and their discourse feature is only used as a means for classifying user utterances.

3. PROPOSED METHOD

We propose using discourse features in addition to acoustic and language model features to train confidence models for slots. Since the slot value is an outcome of exchanges of utterances between the user and system rather than a single utterance, discourse information relevant to the slot is likely to improve the performance of confidence scoring.

We came up with the twelve features enumerated below to express discourse information for slots. They concern the transition of slot values during dialogues and the relationship between current slot values with past user utterances (speech recognition hypotheses) and system utterances. We call the transition of values for a slot the *slot value sequence*. For example, {null → null → Tokyo → Tokyo} is the slot value sequence for *place* in F4 in Fig. 1. Here, the last value Tokyo is the current value. Null means the slot does not have a value.

(D1) Slot purity in slot value sequence: In the slot value sequence, count the times the current value is found, and calculate the ratio of the current value. For example, when the value of the slot *place* changes {Tokyo → Osaka → Kyoto → Osaka}, then the current value Osaka is found in two of the four values, making the slot purity in context 1/2.

(D2) Top slot purity: In the slot value sequence, for all the values that appear, count the number of times each value appears, then calculate the ratio of the value with the highest count. When the value for the slot *place* changes {Tokyo → Osaka → Kyoto → Osaka}, Tokyo, Osaka, and Kyoto are assigned the values of 1/4, 1/2 (2/4) and 1/4, respectively. The maximum value is Osaka's 1/2; therefore, the top slot purity is 1/2.

(D3) Slot variety: Count the number of different values that appear in the slot value sequence. For {Tokyo → Osaka → Kyoto → Osaka}, there are three values "Tokyo, Osaka, Kyoto", and so the slot variety is 3.

(D4) Deny count: Count the number of times the current value has been deleted. For example, consider the sequence {Tokyo → null → Kyoto → Tokyo}. The current value Tokyo is once denied (set to null) by the user (later set to Kyoto). Therefore, the value is 1.

(D5) Overwrite count: Count the number of times the current value has been overwritten by other values. For example, consider the sequence {Tokyo → Osaka → Kyoto → Tokyo}. The current value Tokyo is overwritten once by Osaka. Therefore, the value is 1.

(D6) Continue count: Count the number of times the current value is found in the current slot successively. For example, consider the sequence {null → Tokyo → Tokyo → Tokyo}. Before the current value Tokyo, there are two Tokyo values. Therefore, the value is 2.

(D7) Different value count: Count the number of times the current value is *not* found in the sequence successively. For example, consider the sequence {Tokyo → Osaka → Kyoto → Tokyo}. There are two non-Tokyo values before the current value Tokyo. Therefore, the value is 2.

(D8) Same keyword pair count: According to *Grice's maxim of quantity* [9], which suggests that one has to make one's contribution to the conversation as informative as necessary, a mention of the same slot value to the system's confirmation utterance containing the same value is not desirable. For example, the exchange System: "Are you interested in the weather in Tokyo?" User (recognition hypothesis): "I'm interested in the weather in Tokyo" corresponds to this case. Although the sequence sounds like an implicit confirmation of the system's confirmation request, in terms of Grice's maxim of quantity, it is better for the user to provide more information about his/her intentions. Taking this into account, by looking back at the previous exchanges, we count the number of times the system confirms the current slot value and the user mentions the same value in the next utterance.

(D9) Same keyword count in user utterance: Count the number of times the current value appears in the previous user utterances. For example, when the current value is Tokyo, count the times Tokyo appears in the user utterance history.

(D10) Different keyword count in user utterance: Count the number of times values that are not the current value appear in the previous user utterances. For example, when the current value is Tokyo, count the times non-Tokyo values appear in the user utterance history.

(D11) Same keyword count in system utterance: Count the number of times the current value appears in the previous system utterances. For example, when the current value is Tokyo, count how many times Tokyo appears in the system utterance history.

(D12) Different keyword count in system utterance: Count the number of times values that are not the current value appear in the previous system utterances. For example, when the current value is Tokyo, count the times non-Tokyo values appear in the system utterance history.

4. EXPERIMENT

4.1. System

We prepared a telephone-based spoken dialogue system in the weather information service domain. The system provides Japanwide weather information. Users specify a prefecture name or a city name, a date, and an information type (weather, temperature, precipitation) to obtain the desired information. The speech recognition engine is Julius [10] with its attached acoustic model, and the speech synthesis engine is FinalFluet [11]. The system has a vocabulary of 1,652 words. The language model is a trigram trained from transcriptions obtained from our previous dialogue data collection in the same domain. The system uses a one-best speech recognition hypothesis for understanding. The system has a rule-based dialogue manager and all system utterances are generated by templates.

4.2. Data collection and labeling

We collected dialogue data for confidence model training. Eighteen subjects used the system over the telephone over a period of six days; three subjects per day. Each subject was given a task sheet listing the information to be requested. They were instructed to complete the tasks one by one. Each subject engaged in 16 dialogues, for a total of 288 dialogues collected. Dialogues that took more than three minutes were aborted and regarded as failures. The word error rate (WER) was 40.16%. The task completion rate was 95.83% (276/288). The WER may seem high, but considering the nature of human-computer dialogues in which bad speech recognition prolongs dialogues, the WER here is reasonable. We recorded the system and user utterances and the intention recognition results after each user utterance. The acoustic and language model features and discourse features were extracted for all slot values in the recorded intention recognition results. We hand-labeled the slot values correct or incorrect.

4.3. Data screening

Before training confidence models, we screened the data. First, we discarded the data of slots that did not have values. Then, we removed the data of slots that had just been filled, since they are considered to possess little discourse information. We also removed the data of slots that did not change during the dialogue. Although it might be possible to estimate their confidence from the stillness of the values, we consider it difficult to differentiate *the cases in which values do not change because of repeated misrecognitions* from *those in which the recognizer keeps recognizing the correct values*, because in the data collection, users frequently repeated the same keywords/phrases for emphases and implicit confirmations.

In addition, we did not use the data of *grounded* slots. The system holds a *grounding value*, which is represented by a binary value of *true* or *false*, for each slot indicating whether the value has been acknowledged by the user. For example, when the system confirms by asking "Are you interested in the weather in Tokyo?" and the user says "Yes," then, the grounding values for *info* and *place* are set to *true*. It is natural that slots that have been grounded are basically correct. Therefore, we do not use such data. There were 4812 slot samples in all, and after screening, 777 samples remained (362 positive samples and 415 negative samples).

4.4. Confidence model training

For confidence model training, as acoustic and language model features, we used the same features that Hazen et al. used in [6] (called word-level features in their paper) with some modifications. Modifications had to be made because of the differences in speech recognizers. As discourse features, we used all the discourse features except D10, because we found, after testing several combinations of the features, that it does not have a positive contribution to confidence scoring. The confidence model training method was adopted from [6], which uses a weighted linear combination of features to produce probabilistic confidence scores. The weights were optimized using the training data.

Table 1. False acceptance rate (FAR) and false rejection rate (FRR) for the conventional and proposed models.

	False Acceptance Rate		False Rejection Rate	
	conv.	prop.	conv.	prop.
grouping-1	0.172	0.200	0.385	0.231
grouping-2	0.109	0.124	0.542	0.206
grouping-3	0.432	0.286	0.300	0.333
grouping-4	0.220	0.025	0.339	0.339
grouping-5	0.175	0.077	0.250	0.182
grouping-6	0.286	0.216	0.432	0.341
total	0.218	0.149	0.406	0.257

Table 2. Matrix of counts of correct and incorrect items for the conventional and proposed models.

	prop. correct	prop. incorrect
conv. correct	200	15
conv. incorrect	69	493

4.5. Evaluation

For evaluation, we performed a six-fold cross validation. We first separated the data into six groupings corresponding to the data for the six experiment dates, and trained six confidence models, taking five of the six groupings as training data and making the remaining grouping the test data. For comparison, we also created confidence models that only use acoustic and language model features for training. Hereafter, we call the model trained by acoustic and language model features the *conventional model* (conv. for short), and the model trained by the acoustic and language model features plus the discourse features (w/o D10) the *proposed model* (prop. for short).

Table 1 shows the false acceptance rate (FAR) and false rejection rate (FRR) for the conventional and proposed models when each grouping is used as the test data. The FAR is the rate at which the model incorrectly classifies negative samples as positives, and the FRR the rate at which the model incorrectly classifies positives as negatives. For both FAR and FRR, the proposed model performs better than the conventional model. Table 2 shows the matrix of counts of correct and incorrect items for the conventional and proposed models. Of all the samples, there were 69 that only the proposed model classified correctly, and 15 that only the conventional model classified correctly. From a statistical test (McNemar's test [12]), it was found that the two models have a statistically significant difference in terms of classification performance ($p = 1.94 \cdot 10^{-9}$), which verifies the effectiveness of the discourse features.

4.6. Analysis on the discourse features

We investigated how each of the discourse features affects the classification results. Table 3 shows the F-measure (harmonic mean of

Table 3. F-measure (harmonic mean of precision and recall) for models each trained without D10 and one of the remaining discourse features.

	F-measure	Drop in F-measure
prop. (All w/o D10)	0.794	0.000
w/o D10, D1	0.765	0.029
w/o D10, D2	0.789	0.005
w/o D10, D3	0.782	0.011
w/o D10, D4	0.774	0.020
w/o D10, D5	0.754	0.040
w/o D10, D6	0.787	0.006
w/o D10, D7	0.753	0.041
w/o D10, D8	0.703	**0.091**
w/o D10, D9	0.730	**0.063**
w/o D10, D11	0.773	0.020
w/o D10, D12	0.753	0.040

the precision and recall) for models, each of which was trained without D10 and one of the remaining discourse features. The row indexed by **prop. (All w/o D10)** represents the proposed model and the third column (drop in F-measure) shows the difference of the F-measure from the proposed model.

From the table, one can see that the same keyword pair count (D8) and the same keyword count in user utterance (D9) have relatively larger drop values than the others, indicating that they may be more important than other features. D8 being important may suggest that traditional dialogue theories such as Grice's maxim also stand in human-computer dialogues. As for D9, when we look at its coefficients in the confidence models, we find that the value is positive: the larger the same keyword count, the larger the confidence. This may indicate a strong tendency for users to utter already correctly recognized items many times as implicit confirmations. On the other hand, the top slot purity (D2) has a small drop value. Since the slot purity in slot value sequence (D1) also has a small value, it is suggested that however many times a slot has the same value, the correctness of the slot is not guaranteed.

Fig. 2 shows the FAR-FRR curves for the models without D2, D8, and D9, respectively, along with those for the proposed model and the conventional model. It can be seen clearly that models without D8 and D9 are close to the curve for the conventional model, and the model without D2 is almost on the curve for the proposed model.

5. SUMMARY AND FUTURE WORK

We proposed a confidence scoring method for intention recognition results in spoken dialogue systems. To improve confidence scoring, we utilized both the acoustic and language model features of the speech recognition results and various discourse features related to slot values, such as the number of times that slot values are mentioned in a dialogue. Experimental results show that the proposed method significantly improves the confidence scoring, indicating the effectiveness of the discourse features. Future work will include a further analysis into the impact of the discourse features, handling of slots that we removed this time, an exploration of other discourse features, utilization of the relationships and constraints among the slots, and incorporating confidence scoring results for dialogue management in workable systems.

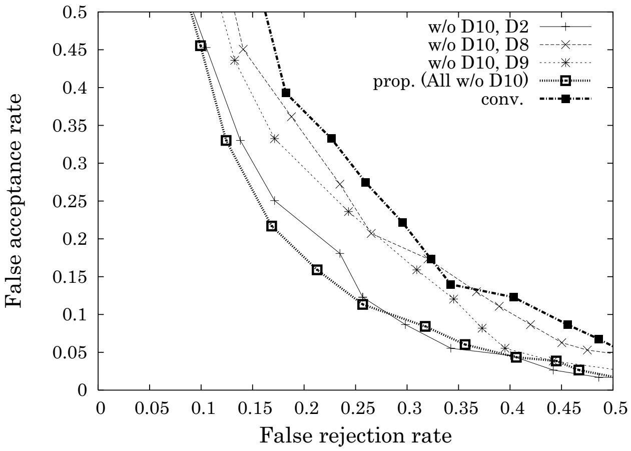

Fig. 2. False Acceptance Rate (FAR) - False Rejection Rate (FRR) curves for the proposed and conventional models and for models that do not use D2, D8, and D9 as discourse features.

6. REFERENCES

[1] Daniel G. Bobrow, Ronald M. Kaplan, Martin Kay, Donald A. Norman, Henry Thompson, and Terry Winograd, "GUS, a frame driven dialog system," *Artif. Intel.*, vol. 8, pp. 155–173, 1977.

[2] Satinder Singh, Diane Litman, Michael Kearns, and Marilyn Walker, "Optimizing dialogue management with reinforcement learning: Experiments with the NJFun system," *Journal of Artificial Intelligence Research*, vol. 16, pp. 105–133, 2002.

[3] Kohji Dohsaka, Norihito Yasuda, and Kiyoaki Aikawa, "Efficient spoken dialogue control depending on the speech recognition rate and system's database," in *Proc. Eurospeech*, 2003, pp. 657–660.

[4] Kazunori Komatani and Tatsuya Kawahara, "Flexible mixed-initiative dialogue management using concept-level confidence measures of speech recognizer output," in *Proc. 18th COLING*, 2000, vol. 1, pp. 467–473.

[5] Frank Wessel, Ralf Schlüter, Klaus Macherey, and Hermann Ney, "Confidence measures for large vocabulary continuous speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 3, pp. 288–298, 2001.

[6] Timothy J. Hazen, Stephanie Seneff, and Joseph Polifroni, "Recognition confidence scoring and its use in speech understanding systems," *Computer Speech and Language*, vol. 16, pp. 49–67, January 2002.

[7] Sameer S. Pradhan and Wayne H. Ward, "Estimating semantic confidence for spoken dialog systems," in *Proc. ICASSP*, 2002, vol. 1, pp. 233–236.

[8] Thomas Schaaf and Thomas Kemp, "Confidence measures for spontaneous speech recognition," in *Proc. ICASSP*, 1997, vol. 2, pp. 875–878.

[9] H. P. Grice, "Logic and conversation," in *Syntax and Semantics 3: Speech Acts*, P. Cole and J. Morgan, Eds., pp. 41–58. New York: Academic Press, 1975.

[10] Akinobu Lee, Tatsuya Kawahara, and Kiyohiro Shikano, "Julius – an open source real-time large vocabulary recognition engine," in *Proc. Eurospeech*, 2001, pp. 1691–1694.

[11] Satoshi Takano, Kimihito Tanaka, Hideyuki Mizuno, Masanobu Abe, and ShiN'ya Nakajima, "A Japanese TTS system based on multiform units and a speech modification algorithm with harmonics reconstruction," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 1, pp. 3–10, 2001.

[12] L. Gillick and S. Cox, "Some statistical issues in the comparison of speech recognition algorithms," in *Proc. ICASSP*, 1989, vol. 1, pp. 532–535.