A METHOD FOR EVALUATING INCREMENTAL UTTERANCE UNDERSTANDING IN SPOKEN DIALOGUE SYSTEMS

Ryuichiro HIGASHINAKA, Noboru MIYAZAKI, Mikio NAKANO, Kiyoaki AIKAWA

NTT Communication Science Laboratories
NTT Corporation
3-1 Morinosato-Wakamiya, Atsugi, Kanagawa 243-0198, Japan
{rh, nmiya, nakano}@atom.brl.ntt.co.jp, aik@idea.brl.ntt.co.jp

ABSTRACT

In single utterance understanding, which does not include discourse understanding, the concept error rate (CER), or the keyword error rate, has been widely used as an evaluation measure for utterance understanding. However, the CER cannot be used for evaluating systems that understand user utterances based on previous user utterances. In this paper, we propose a method for evaluating incremental utterance understanding, which involves speech recognition, language understanding and discourse processing in spoken dialogue systems, by finding a measure that correlates closely with the system's performance based on dialogue states and their way of update. We defined dialogue performance by task completion time, and performed a multiple linear regression analysis using task completion time as the explained variable and various metrics concerning dialogue states as explaining variables. The obtained multiple regression model fits comparatively well and shows validity as an evaluation measure.

1. INTRODUCTION

Due to advances in speech recognition and speech synthesis technologies, spoken dialogue systems have been attracting a lot of attention. There are two types of spoken dialogue systems, those that understand a single user utterance and respond to it without taking context into account, and those that deal with multiple exchanges of utterances by understanding user utterances in the context of dialogues. The latter, which is discussed in this paper, has to be able to appropriately update the dialogue state each time a user utterance is made. Here, the dialogue state is a collection of bits of information that the system internally stores. Included in that information are the understanding result of the user utterances up to that point of time as well as other discourse-related items such as the topic. For correct updating of dialogue states or automatic acquisition of discourse understanding rules, we need to know what kind of dialogue state or sequence of dialogue states contribute to the performance of a spoken dialogue system.

In single utterance understanding, which does not include discourse understanding, the concept error rate (CER), which is also known as the keyword error rate, is widely used. However, the CER cannot be used for systems that work by understanding user utterances based on previous user utterances, because the understanding result might be affected by the previous dialogue state. It is not clear whether the evaluation should focus on the dialogue states themselves or the way they are updated in a dialogue. Currently, there is no metric for evaluating utterance understanding that requires discourse understanding.

Such a metric is especially needed for a system using ISSS (Incremental Sentence Sequence Search) [1], which we introduced. In spoken dialogues, some utterances convey their meaning over several speech intervals. To cope with these, ISSS accepts sentences and sentence fragments (i.e., words, phrases) and incrementally updates the dialogue state. In a way, language understanding and discourse understanding are combined. If ambiguity is found in the understanding of the fragments, ISSS holds multiple contexts ordered by priority and the system can decide on a single context after any speech interval. Since systems that use an ISSS-type method normally understand each utterance based on previous utterances, the importance of discourse understanding is quite high.

This paper proposes a method for evaluating incremental utterance understanding, which involves speech recognition, language understanding, and discourse processing in spoken dialogue systems. The evaluation uses metrics derived from dialogue states and their way of update that correlate closely with system performance. We defined the performance of a dialogue by the task completion time, and performed a multiple linear regression analysis using task completion time as the explained variable and various metrics concerning dialogue states as explaining variables. The next section describes the problem to be solved in detail. After that, various metrics concerning dialogue states are described and then, using our dialogue system, the correlation between these metrics and the task performance is shown. This is followed by the results. The last section summarizes and mentions future work.

2. PROBLEM

Consider a spoken dialogue system that sequentially handles multiple utterances and updates its dialogue state each time it receives a user utterance as shown in Fig. 1. The initial dialogue state (usually void) is changed to dialogue state A by user utterance 1 and then changed again by user utterance 2.

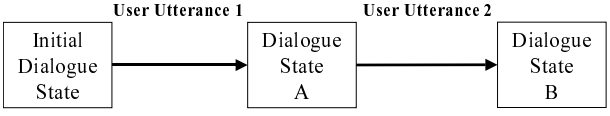

Fig. 1. Dialogue state updates

For systems that always start with the initial dialogue state, the CER is suitable for evaluating utterance understanding. However, for systems which incrementally update dialogue states as in the figure, the CER cannot be used, because it does not reflect previous dialogue states. For example, dialogue state B cannot be derived when the initial dialogue state gets the same user utterance, utterance 2. In CER-usable systems, the correct dialogue state after an utterance is clear. However, when we take previous dialogue states into account, it becomes unclear what the correct dialogue state is. For example, dialogue state B might be wrong as a resulting state, but it may have been updated correctly in part. In an ISSS-based system, in which language understanding and discourse understanding are combined, the importance of discourse understanding is very high. For example, an utterance like "from three" (pause) "to four" is processed as in Fig. 2. It shows the case when "from three" was misrecognized as "from two". As a result, the wrongly updated dialogue state affected the subsequent dialogue state.

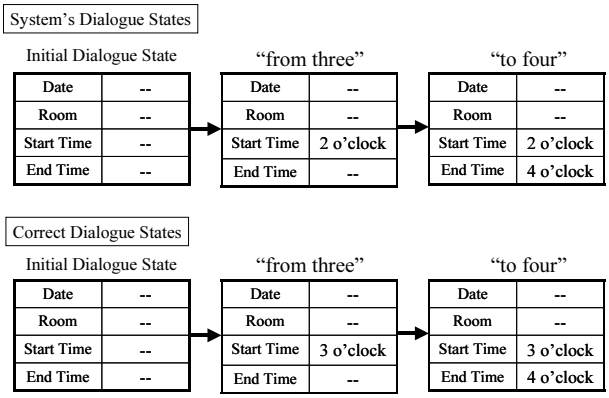

Fig. 2. Example of dialogue state updates

To evaluate such systems, we need a measure for evaluating incremental utterance understanding. If we have such a measure, we will be able to build spoken dialogue systems having good incremental utterance understanding, use it for automatically acquiring discourse understanding rules, and test those rules using simulations without performing costly dialogue data collection. Currently, it is not clear whether the evaluation should focus on the dialogue states themselves or the way they are updated in a dialogue.

3. APPROACH

To tackle this problem, we enumerate possible metrics concerning dialogue states and choose those that have good correlation with the system's performance.

3.1. Metrics Concerning Dialogue States

In spoken dialogue systems in which discourse understanding plays a crucial role, a dialogue state at a certain point of time is, after receiving a user utterance, updated to the next dialogue state based on discourse understanding rules. This updating from one state to another is called an understanding unit. We assume that a dialogue state is expressed as a frame expression, which is common in many systems [2]. A frame is a bundle of slots that consist of attribute-value pairs concerning a certain domain. The initial dialogue state in an understanding unit is called the initial frame, and that after understanding, the final frame. Based on the final frame, the system makes responses. A dialogue consists of a sequence of understanding units and system responses. The aim of this research is to discover what kind of dialogue state or sequence of dialogue states contribute to the performance of a spoken dialogue system. Since a dialogue comprises multiple understanding units, we represent a dialogue state in a single understanding unit, and then, for the whole dialogue, we use their average. Below is the description of how to represent a dialogue state in an understanding unit. Let the system's dialogue state after receiving user utterance be a hypothesis frame, and the ideal dialogue state, which needs to be hand-crafted, a reference frame. The representation of a dialogue state is derived by comparing the hypothesis frame and the reference frame.

The comparison is performed in two ways. One is a simple comparison of each value of their slots, to see if the values are the same or different or if the slots have values at all. From this comparison, each slot of a hypothesis frame is given one of four labels (see Table 1). In the table, the hypothesis frame is written as Hyp and the reference frame as Ref.

label	name	description
C	Correct	Ref and Hyp has the same value.
I	Insertion	Ref does not have a value, but Hyp has a value.
D	Deletion	Ref has a value, but Hyp does not have a value.
S	Substitution	Ref and Hyp both have different values.

Table 1. Labels given to each slot of a hypothesis frame

The other comparison is performed using changes from the initial frame, namely, "the difference between the initial frame and the hypothesis frame" is compared with "the difference between the initial frame and the reference frame". From this comparison, five types of labels are given to each slot of a hypothesis frame (see Table 2).

label	name	description
CU	Correct Update	Ref and Hyp both change to the same value.
CL	Correctly Left	Ref and Hyp both do not change correctly.
UD	Update Deletion	Ref changes, but Hyp does not change.
UI	Update Insertion	Ref does not change, but Hyp changes.
US	Update Substitution	Ref and Hyp both change to different values.

Table 2. Labels given to the change of each slot of a hypothesis frame

From these nine types of labels given to each slot of a hypothesis

frame, we derive ten metrics for representing a dialogue state[1]:

1. **slot accuracy**
$$\frac{C}{number\ of\ slots}$$
2. **insertion error rate**
$$\frac{I}{number\ of\ slots}$$
3. **deletion error rate**
$$\frac{D}{number\ of\ slots}$$
4. **substitution error rate**
$$\frac{S}{number\ of\ slots}$$
5. **slot error rate**
$$\frac{sum\ of\ error\ slots}{number\ of\ slots} = \frac{I+D+S}{number\ of\ slots}$$
6. **update precision**
$$\frac{number\ of\ correctly\ changed\ slots}{number\ of\ changed\ slots\ in\ Hyp} = \frac{CU}{CU+US+UI}$$
7. **update insertion error rate**
$$\frac{number\ of\ changed\ slots\ in\ Hyp}{number\ of\ unchanged\ slots\ in\ Ref} = \frac{UI}{CL+UI}$$
8. **update deletion error rate**
$$\frac{number\ of\ unchanged\ slots\ in\ Hyp}{number\ of\ changed\ slots\ in\ Ref} = \frac{UD}{CU+US+UD}$$
9. **update substitution error rate**
$$\frac{number\ of\ incorrectly\ changed\ slots\ in\ Hyp}{number\ of\ changed\ slots\ in\ Ref} = \frac{US}{CU+US+UD}$$
10. **speech understanding rate**
$$\frac{number\ of\ intervals\ of\ perfect\ slot\ accuracy}{number\ of\ speech\ intervals}$$

These metrics represent the dialogue state of an understanding unit. In this paper, the state of a whole dialogue is represented by the average of each metric, namely, by dividing each metric by the number of speech intervals, excluding the speech understanding rate.

3.2. Performance Measure

In this research, the aim of a dialogue is to complete a task. Therefore, task completion time is used to represent the performance of a dialogue. Though user satisfaction is not directly taken into account in this paper, there is a report suggesting that task completion time correlates closely with user satisfaction [3]. The task completion time can be influenced by the action of the dialogue management component of a system, which we usually call a dialogue strategy. To focus only on dialogue states and dialogue performance, it is necessary to prepare more than one strategy to absorb that influence. Task completion time can also be influenced by the task content (e.g., rooms, dates, and time for reservation in the meeting room reservation domain). Therefore, task completion time should be normalized using task and dialogue strategy.

[1]Labels C,I,D,S,CU,CL,UD,UI,US in the metrics represent the number of slots labeled respectively.

4. EXPERIMENT

4.1. Data Collection

To investigate whether the ten metrics have any correlation with the dialogue performance, we collected dialogue data for analysis. The dialogue data was collected on naive users in acoustically insulated booths. The spoken dialogue system used was developed using the spoken dialogue system toolkit WIT [4]. The domain was meeting room reservation. Subjects were instructed to reserve one or two meeting rooms on one or two dates from a certain time to a certain time. We prepared five task patterns. As a speech recognition engine, we used Julius3.1 [5] with its attached acoustic model. For the language model, we made N-gram from randomly generated texts of acceptable phrases. For system response, NTT's speech synthesis engine Final Fluet [6] was used.

The system has a vocabulary of 160 words, each registered with a category and a semantic feature in its lexicon. There are 18 rules for lexical analysis and 38 rules for parsing and discourse processing. The dialogue state is represented with a frame representation comprising six domain-dependent slots. The system also has three other slots that contain information about the current topic and flags indicating what has been confirmed. We only used domain-dependent slots for analysis, because of difficulty and ambiguity of labelling topics and flags. As described in the previous section, more than one strategy is needed. We prepared two: One is that the system accepts user utterances, until it has enough information to complete the reservation or the user explicitly requests a system response. The other confirms each user utterance. We recorded system's utterances, start and end times of user's utterances and frames before and after the user utterance. The user's voice and system's voice were also recorded, and all user utterances were transcribed.

One subject performed ten dialogues (five tasks on two dialogue strategies). We collected 180 dialogues from 18 subjects (9 males and 9 females). The system recognized and processed 3595 speech intervals in total, excluding barge-ins. Dialogues that took more than five minutes were regarded as failure. The task completion rate was 63.6% (112/176)[2]. We did not use, for analysis, unsuccessful dialogues, whose task completion times were not available.

4.2. Reference Frame

In order to obtain reference frames, a large hand-labeling effort is necessary. Therefore, we first made a simulation system that receives an initial frame and a transcribed text as input and outputs a final frame. Then, human labellers corrected those simulated final frames to make them suitable as reference frames. In this way, the labelling effort was greatly reduced.

4.3. Results

We used the 108 dialogue data that remained after we removed those which had inconsistent logs with transcribed texts. We performed a multiple linear regression analysis using task completion time normalized by the task and the dialogue strategy as the explained variable (Y) and the ten metrics as explaining variables. By stepwise regression, seven metrics were incorporated as a result.

[2]Four dialogues were removed as outliers due to system inadequacies.

The resulting equation is

$$Y = -4.19 - 12.49x_1 + 12.77x_2 - 0.03x_3 - 17.74x_4 + 4.54x_5 + 2.11x_6 + 2.98x_7 \quad (1)$$

where x_1 is the insertion error rate, x_2 the substitution error rate, x_3 the update precision, x_4 the update insertion error rate, x_5 the update deletion error rate, x_6 the update substitution error rate, and x_7 the speech understanding rate.

RSquare is 0.57 and the RSquare Adjusted is 0.54. RMSE (Root Mean Square Error) is 0.63. The model fits comparatively well and shows validity as an evaluation measure. The distribution of actual and predicted task completion times is shown in Fig. 3.

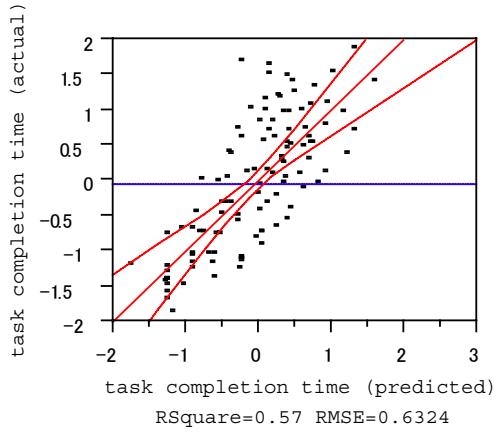

Fig. 3. Distribution of actual and predicted task completion times

The correlation coefficients of the ten metrics against task completion time are shown in Table 3. The update deletion error rate has a relatively high correlation with correlation coefficient 0.62 followed by -0.45 of update precision.

	task completion time
slot accuracy	−0.40
insertion error rate	−0.07
deletion error rate	0.29
substitution error rate	0.40
slot error rate	0.40
update precision	−0.45
update insertion error rate	0.15
update deletion error rate	0.62
update substitution error rate	0.24
speech understanding rate	−0.42

Table 3. Correlation coefficients of the ten metrics against task completion time

5. SUMMARY AND FUTURE WORK

This paper proposes a method for evaluating incremental utterance understanding, which involves speech recognition, language understanding, and discourse processing in spoken dialogue systems, by performing a multiple linear regression analysis using task completion time as the explained variable and various metrics concerning dialogue states as explaining variables. The obtained multiple regression model fits comparatively well and shows validity as an evaluation measure. It also indicates that we should take into account both the dialogue states themselves and their way of update.

There are some issues remaining. For example, the domain used in this paper is small. It is not clear whether we will get the same results in bigger domains, such as flight reservation. The low motivation of subjects due to the lack of reality of a domain is also something we have to consider. We also need to consider user satisfaction and other metrics concerning dialogue states. Despite the above limitations, the result of the experiment suggests that our approach is promising for evaluating incremental utterance understanding.

6. ACKNOWLEDGEMENTS

We thank Dr. Hiroshi Murase and all members of the Dialogue Understanding Research Group for helpful comments. We also thank Atsushi Fukayama for advise on the multiple linear regression analysis.

7. REFERENCES

[1] M. Nakano, N. Miyazaki, J. Hirasawa, K. Dohsaka, and T. Kawabata, "Understanding unsegmented user utterances in real-time spoken dialogue systems," in *Proc. 37th ACL*, 1999, pp. 200–207.

[2] D. G. Bobrow, R. M. Kaplan, M. Kay, D. A. Norman, H. Thompson, and T. Winograd, "GUS, a frame driven dialog system," *Artif. Intel.*, 8:155–173, 1977.

[3] M. Walker, C. Kamm, and D. Litman, "Towards developing general models of usability with paradise," *Natural Language Engineering: Special Issue on Best Practice in Spoken Dialogue Systems.*, 2000.

[4] M. Nakano, N. Miyazaki, N. Yasuda, A. Sugiyama, J. Hirasawa, K. Dohsaka, , and K. Aikawa, "WIT: A toolkit for building robust and real-time spoken dialogue systems," in *Proc. SIGDIAL*, 2000, pp. 150–159.

[5] A. Lee, T. Kawahara, and K. Shikano, "Julius – an open source real-time large vocabulary recognition engine," in *Proc. Eurospeech*, 2001, pp. 1691–1694.

[6] S. Takano, K. Tanaka, H. Mizuno, M. Abe, and S. Nakajima, "A Japanese TTS Syntem Based on Multi-form Units and a Speech Modification Algorithm with Harmonics Reconstruction," *IEEE Transactions on Speech and Processing*, 9(1):3–10, 2001.