A Spoken Dialog System with Redundant Response to Prevent User Misunderstanding

Masaki Yamaoka*, Sunao Hara* and Masanobu Abe*
* Okayama University, Okayama, Japan
E-mail: yamaoka@a.cs.okayama-u.ac.jp, hara@okayama-u.ac.jp, abe@cs.okayama-u.ac.jp

***Abstract*—We propose a spoken dialog strategy for car navigation systems to facilitate safe driving. To drive safely, drivers need to concentrate on their driving; however, their concentration may be disrupted due to disagreement with their spoken dialog system. Therefore, we need to solve the problems of user misunderstandings as well as misunderstanding of spoken dialog systems. For this purpose, we introduced a driver workload level in spoken dialog management in order to prevent user misunderstandings. A key strategy of the dialog management is to make speech redundant if the driver's workload is too high in assuming that the user probably misunderstand the system utterance under such a condition. An experiment was conducted to compare performances of the proposed method and a conventional method using a user simulator. The simulator is developed under the assumption of two types of drivers: an experienced driver model and a novice driver model. Experimental results showed that the proposed strategies achieved better performance than the conventional one for task completion time, task completion rate, and user's positive speech rate. In particular, these performance differences are greater for novice users than for experienced users.**

I. Introduction

Spoken dialog systems (SDSs) are now commonly used because of performance improvements in automatic speech recognition systems and speech synthesis systems. An obvious example of using SDSs is in location finding with car navigation systems. SDSs enable drivers to use such systems in hands-free mode, keeping their eyes on the road. Drivers have to perform two tasks, which are driving the car and talking to the system, simultaneously. Lee *et al.* [1] pointed out the problem that the driver may deflect their attention away from the driving task if there is a misunderstanding with the SDS. This fact has the potential risk of inattentive driving that may cause car crashes or traffic accidents. Some aspects of safe driving should be introduced to develop advanced SDSs for car navigation systems.

The risk of accidents may decrease as the accuracy of speech understanding improves. In this aspects, there are several researches into spoken dialog strategies to become robust to SDS misunderstanding. Moreover, as another aspect of safe driving, we focus on the driver's workload levels while driving. The SDSs should not prevent drivers' attentions by its response, especially, if the drivers' workload levels are high. There are various methods for estimating a driver's workload, for example, using a large number of automotive sensors [2], [3] and speech-related features [4]. However, there are fewer researches into the strategies using the drivers' workload levels.

In this paper, we propose a novel dialog strategy which can provide robustness to both the misunderstanding and safe driving problems with SDSs. A key idea is to introduce drivers' workload aspects in a graph-based search algorithm. The graph search keeps multiple understanding hypotheses in the information retrieval process [5]; therefore, it can achieve robustness to system misunderstanding. Safety is assured by introducing the drivers' conditions to the graph search algorithms.

An evaluation experiment with a user simulator is conducted to consider whether the proposed method is effective. User simulator make possible to try out the evaluations with various but reproducible conditions. Naturally, it also make free from worry of traffic accident during the experiment. In this study, we implemented a user simulator which simulates two types of users: experienced drivers and novice drivers.

II. SDS for car navigation systems

First, we define a task of our system in the section A. Next, we mention about the baseline algorithms for a speech understanding and a dialog management in the next sections B and C. Finally, we describe our proposed method in the section D.

A. Location finding task

Our system is able to find a destination that will be visited en route to the main destination while driving. There are three types of destinations: parking, convenience store, and family restaurant. The tasks of "parking" and "family restaurant" find the cheapest place that matches the query, while "convenience store" finds the nearest. The slots of each destination are given as below.

Parking
"Furthest distance limit", "expected parking time"

Convenience store
"Furthest distance limit"

Family restaurant
"Furthest distance limit", "price range"

The major dialog actions of the system are questions for obtaining new information or confirmation for uncertain information. Additionally, we propose adding guidance speech before the questions. We assume that drivers will mishear the system speech if they are spoken by the system under high workload conditions, e.g., turning at intersections, driving in

[Low workload:Don't add guidance]
sys : Tell me what kind of destination.
usr : Parking.
sys : Is the destination parking?
usr : Yes.
[High workload:Add guidance]
sys : I want to ask you about an upper limit for distance.
sys : Give me an upper limit for distance.
usr : 2 km.
[Low workload:Don't add guidance]
sys : Tell me how long you wish to park.
usr : 3 hours.
sys : There is a parking nearby.
sys : Shall I set it as the destination?
usr : Yes.

Fig. 1: An example of a dialog

TABLE I: A detail of system responses

Questions for new information:
"Tell me what kind of destination."
"Give me an upper limit for distance."
"Tell me how long you wish to park."
"Give me a price range."
Questions for confirmation:
"The kind of destination is "DESTINATION." Is this right?"
"The furthest distance is "DISTANCE." Is this right?"
"The expected parking time is "TIME." Is this right?"
"The price range is "PRICE." Is this right?"
Guidance:
"I want to ask you about the type of destination."
"I want to ask you about an upper limit for distance."
"I want to ask you about the expected length of parking time."
"I want to ask you about a price range."

a traffic jam, etc. The guidance speech readies users to hear the system speech, preventing misunderstanding. Example of the dialog with the system is shown in Figure 1. The details of the questions and guidance dialog are shown in Table I

B. Speech understanding on the basis of graph search algorithms

The speech understanding (SU) algorithm of the system is based on a graph search algorithm on the spoken dialog [5]. The dialog strategy is slot-filling, but its filling procedure is optimized for the graph search algorithm. A search graph of SU is constructed by considering a keyword set as a node of a graph. Active nodes are the current understanding status. The system expands the nodes based on speech recognition results obtained from each dialog step. By allowing a number of active nodes to be present, it is possible to keep a number of understanding hypotheses. If an incorrect search occurs, it is easy to reach correct understanding by backtracking.

The selection of the system response is realized by a best first-search algorithm. In general, the best-first search is a method to find which node should be expanded in the next expansion step. In considering SU, the best score node means the best SU state in the SU tree; therefore, the system should generate new responses which are appropriate to expand the best score node. Each node has a score which is the sum of the confidence scores included in the nodes. The confidence score is given by an automatic speech recognizer.

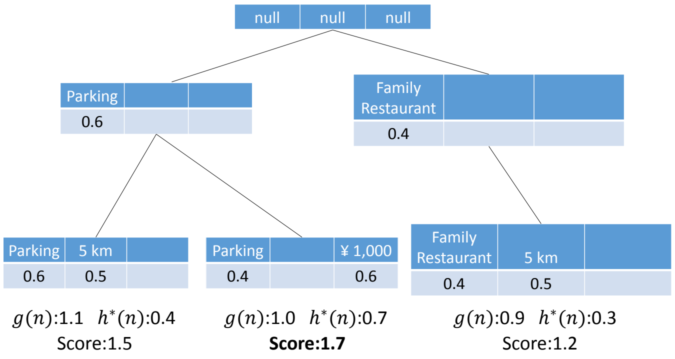


Fig. 2: *Best-first search with heuristics.*

This score is referred to as a search score $g(n)$ in the search process. We adopt a heuristic score $\hat{h}(n)$ to the best first search, as shown in Figure 2. The highest score $g(n) + \hat{h}(n)$ node is selected. The system's speeches are selected based on the keyword of the node, and nodes are expanded using speech recognition results.

C. Baseline algorithm for a dialog management

Kitaoka *et al.* [5] used consistency measure $S_c(q)$ and efficiency measure $S_e(q)$ to calculate the heuristic score $\hat{h}(n)$. This is expected to make the dialog for the search efficient and consistent. The searching method is the following.

First, the system select a node to make a response based on its current graph and a user response.

$$\hat{n} = \underset{n}{\operatorname{argmax}}\{g(n) + \hat{h}(n)\} \tag{1}$$

where $\hat{n}$ is a node that is expanded.

$\hat{h}(n)$ is the following.

$$\hat{h}(n) = w_c S_c(\hat{q}) + w_e S_e(\hat{q}) \tag{2}$$

The consistency measure is as below:

$$S_c(q) = 1 - I(q, n) \tag{3}$$

$I(q, n) = 1$ when question q conflicts with n, and $I(q, n) = 0$ otherwise. Then, the efficiency measure is as below:

$$S_e(q) = \left\{ N(n) - \frac{1}{|A_q|} \sum_{a \in A_q}^{|A_q|} N(q, a, n) \right\} \Big/ N(n) \tag{4}$$

where A_q is a set of possible answers given by the user after asking question q. $N(q, a, n)$ is the number of retrieval candidates, answered by the user by after asking question q. $N(n)$ is the number of retrieval candidates with node n. If the number of retrieval candidates is 0, $N(q, a, n) = N(n)$.

Finally, the system selects question $\hat{q}$ with its maximum weighted sum.

$$\hat{q} = \underset{q}{\operatorname{argmax}}\{w_c S_c(q) + w_e S_e(q)\} \tag{5}$$

D. Algorithm extension introducing redundant response

In this study, we introduce a redundancy measure $S_r(q)$ to use the SDS following safe driving practices. When the driver's workload is low a short utterance is better, while a long utterance is better when the workload is high because this is when the driver may misunderstand. In this paper, if the system selects an utterance with guidance, this utterance is redundant, otherwise it is not redundant. Thus, we use the following.

$$S_r(q) = guid(q) \tag{6}$$

where $guid(q) = 1$ when selecting question with guidance q, and $guid(q) = 0$ otherwise.

Then, the equations (2) and (5) is extended as below:

$$\hat{h}(n) = w_c S_c(\hat{q}) + w_e S_e(\hat{q}) + w_r S_r(\hat{q}) \tag{7}$$

$$\hat{q} = \underset{q}{\operatorname{argmax}}\{w_c S_c(q) + w_e S_e(q) + w_r S_r(q)\} \tag{8}$$

We set $w_c = w_e = 0.5$, where w_r is the driver's workload value that is calculated from various sensors.

III. User simulator

An actual driver is affected by various factors, for example, road conditions, car model, and the user themselves. All of these factors, we focus on the user. We consider that changes in the driver's workload have several patterns. The user actions are specific to the driver's workload patterns. For this reason, we express changes of the driver's workload by automaton, and we use two user models. The driver's workload model using automaton is shown in Figure 3. We can express various user models by changing parameter, p_1, p_2, q_1, and q_2.

IV. Evaluation

A. Experimental conditions

An experiment using the dialog simulator is conducted to compare performance with two methods: the proposed method with redundant response and the baseline method without redundant response. For the comparison purpose, we also evaluated the baseline method which is driven under the ideal condition, that is, a user will never misrecognized the system response. Thousand simulations are done for each method for each value of recognition rate. We define recognition rate between 60%–100%. Speech rate is 8 mora per second, the system and the user speech interval is 1 second.

In this experiment, we assume that there are two types of model; i.e., experienced driver and novice driver. The former is $p_1 = 0.5, q_1 = 0.5, p_2 = 0.8, q_2 = 0.2$. The latter is $p_1 = 0.5, q_1 = 0.5, p_2 = 0.2, q_2 = 0.8$.

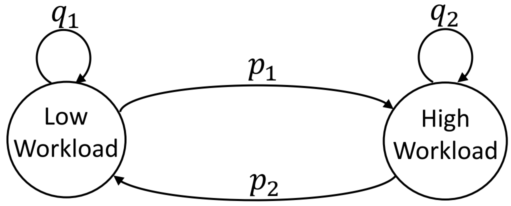


Fig. 3: *The driver's workload model representing the automaton model.*

We assume that if the system communicates with the driver without guidance and the driver's workload is high, then there is an 80% probability of misunderstanding of the content by the driver, and this causes them to ask content-related questions. The driver's workload value has a range from 0 to 1, and is used as w_r in equations (7) and (8). If the driver is under the low workload state, the value is set between 0 and 0.4 at random and if the driver is under the high workload state, the value is set between 0.5 and 1.0 at random.

B. Performance criteria

The task is successful when the system recommends the user-intended destination and is deemed to have failed when the system recommends an unintended destination or the dialog takes a long time. In this paper, we use 400 s for the time.

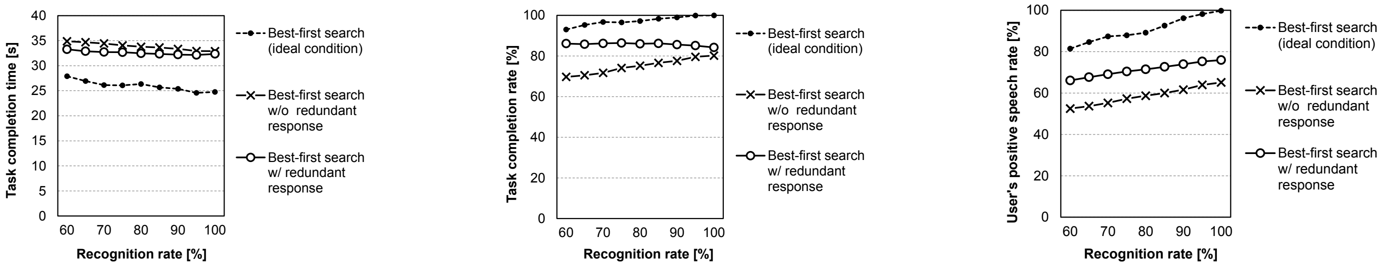
We evaluate dialog performances by three criteria which are given as below:

1) Task completion time
This measure is the amount of time spent until the task is successfully completed. The dialog strategy which spends the shortest time is the better one.
2) Task completion rate
This measure means the ratio of the number of successfully completed tasks to the number of all tasks. The dialog strategy which has the higher rate is the better one.
3) Positive speech rate
This measure is the ratio of the number of a user's non-negative responses to the number of all turns. A non-negative response means not only positive words, "yes," but also value specific words, e.g., "2 km." The higher this rate, the smoother is the dialog.

C. Simulation results

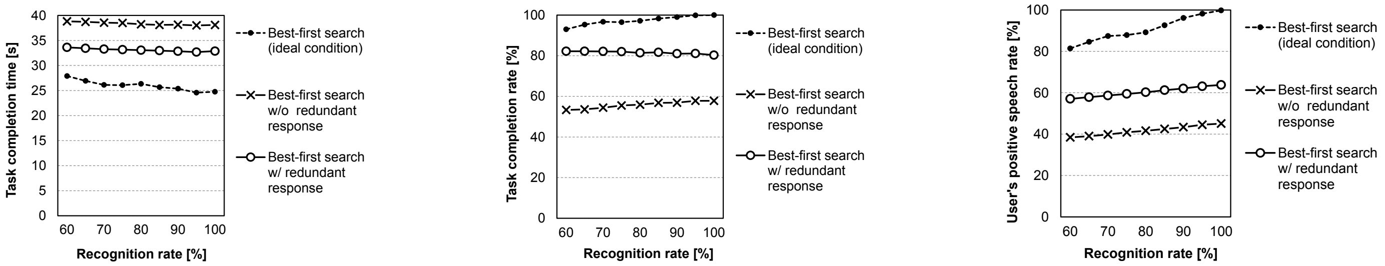
First, the task completion times of the experienced driver model and novice driver model are shown in Figures 4(a) and 5(a). The task completion time is shorter in the case of best-first search with redundant response in both user models because the number of times asking content-related question decreases using redundant responses. There is a big difference between using and not using best-first search using redundant response in the novice driver model. In contrast, there is not much difference between using and not it in the experienced driver model because the question-asking content increases with not using the redundant responses.

Next, the task completion rate of the experienced driver model and novice driver model are shown in Figures 4(b) and 5(b). As with task completion time, task completion rate shows good results in the case of the best-first search using redundant response in both user models. If the recognition rate increases, the task completion rate increases in the case of not using redundant response, but not in the case of the proposed strategies. This phenomenon is salient in the result of experienced driver model. These results suggest that the proposed



(a) *Task completion time* (b) *Task completion rate* (c) *Positive speech rate*

Fig. 4: *Evaluation results by simulation of experienced driver*



(a) *Task completion time* (b) *Task completion rate* (c) *Positive speech rate*

Fig. 5: *Evaluation results by simulation of novice driver*

system can achieve the stably higher performance regardless of both experience level of users and the performance of the speech recognition system.

Finally, the positive speech rates of the experienced driver model and novice driver model are shown in Figures 4(c) and 5(c). As with task completion time, the positive speech rate is shorter in the case of the best-first search using redundant response in both user models. There is a big difference between using and not using best-first search using redundant responses in the novice driver model. In contrast, there is not much difference between using and not using it in the experienced driver model.

From these results, the best-first search with redundant response is an effective strategy. The result also indicated that the proposed method will make good user experience for the novice users. It is important factor for novel systems because there is no experienced user from the beginning.

V. Conclusions

In this paper, we proposed a strategy that includes redundant speech to achieve safe driving in a car. The strategy was based on a graph search algorithm to select an appropriate system response. The criteria of each node in the graph were calculated from the aspects of efficiency, consistency, and redundancy. We carried out a performance evaluation experiment to compare with a conventional strategy. Experimental results showed that this strategy achieved better performance than the conventional strategy from every aspect, particularly with regard to redundant speech being effective for beginner drivers.

Effectiveness of the proposed method is suggested by the experiments, but some future works are still remaining. The detail of the relationship between a driver's workload and the frequency of misunderstanding should be investigated. The trade-off between driver's workload and dialog performance when the system selects redundant speech should be also investigated. The research of cognitive load [6] might be important for a definition of driver's workloads in our experiments. We also have to develop a measuring system of driver's workload from several sensor data, and evaluate the effectiveness of the proposed method in a real driving environment. The differences of the result of this paper and a result in a real driving environment are also interesting research topics.

VI. Acknowledgments

This work was supported by JSPS KAKENHI 26730092.

References

[1] J. D. Lee, B. Caven, S. Haake, and T. L. Brown, “Speech-based interaction with in-vehicle computers: The effect of speech-based e-mail on drivers' attention to the roadway,” *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 43, no. 4, pp. 631–640, 2001.

[2] L. Malta, C. Miyajima, N, Kitaoka, and K. Takeda, “Analysis of real-world driver's frustration,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 1, pp. 109–118, 2011.

[3] Y. Uchiyama, S. Kojima, T. Hongo, R. Terashima, and T. Wakita, “Voice information system adapted to driver's mental workload,” *Proc. of the Human Factors and Ergonomics Society Annual Meeting*, vol. 46, no. 22, pp. 1871–1875, 2002.

[4] T. Kleinshmidt, P. Boyraz, H. Boril, S. Sridharan, and J. H. L. Hansen, “Assessment of speech dialog systems using multi-modal cognitive load analysis and driving performance metrics,” *ICVES'09*, pp. 162–167, 2009.

[5] N. Kitaoka, Y. Kinoshita, S. Hara, C. Miyajima, and K. Takeda, “A graph-based spoken dialog strategy utilizing multiple understanding hypotheses,” *Information and Media Technologies*, vol. 9, no. 1, pp. 111–120, 2014.

[6] S.L. Oviatt, “Human-centered design meets cognitive load theory: designing interfaces that help people think,” *Proc. of the 14th annual ACM international conference on multimedia*, pp. 871–880, 2006.