Modeling Spoken Decision Making Dialogue and Optimization of its Dialogue Strategy

TERUHISA MISU, KOMEI SUGIURA, KIYONORI OHTAKE, CHIORI HORI, HIDEKI KASHIOKA, HISASHI KAWAI and SATOSHI NAKAMURA
National Institute of Information and Communications Technology
Kyoto, Japan

This paper addresses a user model for user simulation in spoken decision-making dialogue systems. When selecting from a set of alternatives, users have various decision criteria for making decision. Users often do not have a definite goal or criteria for selection, and thus they may find not only what kind of information the system can provide but their own preference or factors that they should emphasize. In this paper, we present a user model and dialogue state expression that consider user’s knowledge and preferences in spoken decision-making dialogue. In order to estimate the parameters of the user model, we implement a trial sightseeing guidance system and collected dialogue data. Then, we model the dialogue as partially observable Markov decision process (POMDP), and optimize its dialogue strategy so that users can make a better choice.

Categories and Subject Descriptors: I.2.8 [Artificial Intelligence]: Problem Solving, Control Methods, and Search—Control theory; H.5.2 [Information Interfaces and Presentation]: User Interfaces—Natural Language

General Terms: Languages, Management

Additional Key Words and Phrases: Spoken dialogue systems, dialogue management, reinforcement learning, decision support systems

1. INTRODUCTION

Over the years, a significant number of spoken dialogue systems have been developed. Their typical task domains include airline information (ATIS & DARPA Communicator) [Levin et al. 2000; Potamianos et al. 2000; Rudnicky. et al. 2000; Seneff and Polifroni 2000] and railway information [Lamel et al. 1999; Sturm et al. 1999; Lamel et al. 2002]. Dialogue systems, in most cases, are used in the fields of database (DB) retrieval [Zue et al. 2000; Komatani et al. 2005; Raux et al. 2005] and transaction processing [Bohus et al. 2006; Pinault et al. 2009], and dialogue strategies are optimized so as to minimize the cost of information access. (e.g., the number of turns [Dohsaka et al. 2003])

In many situations where spoken dialogue interfaces are installed, information...
access by the user is not a goal in itself, but a means for a decision making [Polifroni and Walker 2008]. For example, in using a restaurant retrieval system, the user’s goal may not be the extraction of price information but to make a decision based on the retrieved information on candidate restaurants.

This work focuses on user modeling in such consulting dialogue, when he/she does not have enough knowledge about the target domain, and thus use a (information-kiosk like) system. In such a situation, users are often unaware of not only what kind of information the system can provide but their own preference or factors that they should emphasize. In that sense, the situation of this work is regarded as a kind of the exploratory search [Marchionini 2006]. The system, too, has little knowledge about the user, or where his/her interests lie. Thus, the system has to bridge such gaps by sensing (potential) preferences of the user and recommend information that the user would be interested in considering a trade-off with the length of the dialogue [Lemon 2008].

In this paper, we propose a user model and dialogue state expression that considers the user’s preferences as well as his/her knowledge about the domain changing through a decision making dialogue. A user simulator is trained using a statistics collected using a trial sightseeing guidance system. Next, we optimize the dialogue strategy of the system via reinforcement learning.

The paper is organized as follows. Section 2 gives an overview of spoken decision making dialogue. Section 3 an overview of the system and task domain. Section 4 reports implementation and trail test of the dialogue system and explain why optimization is needed. Section 5 describes the proposed user model and dialogue state expression. We then introduce an online learning method using these models by reinforcement learning. Section 6 concludes the paper.

2. SPOKEN DECISION MAKING DIALOGUE

2.1 Decision support based on information retrieval and navigation

We assume a situation where a user selects from a given set of alternatives. This is highly likely in real world situations; for example, the situation wherein a user selects one restaurant from a list of candidates presented by a car navigation system. In this work, we deal with a sightseeing planning task where the user determines the sightseeing spot to visit, with little prior knowledge about the target domain. We have investigated human-human dialogue with such a task, and reported that such consulting usually consists of a sequence of information requests from the user, presentation and elaboration of information about certain spots by the guide to explain why the spot is recommended followed by the user’s evaluation [Ohtake et al. 2009; Misu et al. 2009]. We thus focus on these interactions.

2.2 Decision support dialogue system

Such consulting dialogues are also regarded as a type of decision making problem. That is, the user selects an alternative from a given set of alternatives based on some criteria. Several studies have featured decision support systems in the operations research field, and the typical method that has been employed is the Analytic Hierarchy Process [Saaty 1980] (AHP). In the AHP, the problem is modeled as a hierarchy that consists of the decision goal, the alternatives for achieving it, and
Preparing Articles for the ACM Transactions

Choose the optimal spot

1. Cherry Blossoms
2. Japanese Garden
3. Easy Access

Kinkakuji Temple
Ryoanji Temple
Nanzenji Temple

Fig. 1. Hierarchy structure for sightseeing guidance dialogue

the criteria for evaluating these alternatives. In the case of the sightseeing guidance task, the goal is to decide on an optimal spot, which is in agreement with the user’s preference. The alternatives include all sightseeing spots that can be proposed and explained by the system. As the criteria, we adopt the “determinants” that are defined in our tagging scheme [Ohtake et al. 2009]. The determinants include various factors that are used to plan sightseeing activities, such as “cherry blossoms” and “Japanese garden.” An example hierarchy using these criteria is shown in Figure 1. We assume that the number of alternatives is relatively small and that the names of all alternatives are known to the users.

In this model, to the user, the problem of making an optimal decision can be solved by fixing a weight vector \( \mathbf{P}_{\text{user}} = (p_1, p_2, \ldots, p_M) \) for criteria and local weight matrix \( \mathbf{V}_{\text{user}} = (v_{11}, v_{12}, \ldots, v_{1M}, \ldots, v_{NM}) \) for alternatives in terms of the criteria. The optimal alternative is then identified by selecting the spot \( k \) with the maximum priority of \( \sum_{m=1}^{M} p_m v_{km} \). In typical AHP methods, the procedure of fixing these weights is often conducted through pairwise comparisons for all the possible combinations of criteria and spots in terms of the criteria, followed by weight tuning based on the results of the comparisons [Saaty 1980]. However, the methodology cannot be directly applied to spoken dialogue systems. To users, the information about the spot in terms of the criteria is not known, but is obtained only through the system’s information navigation. Thus, it is difficult to evaluate and compare the spots without the system’s information navigation. In addition, spoken dialogue systems usually handle several candidates and criteria, making pairwise comparison a costly affair. Although there have been several studies that are dealing with decision making with spoken dialogue interface [Polifroni and Walker 2008], these works usually assume that the users know all criteria that the users/system can use for making decision. We thus consider a spoken dialogue framework that estimates the weights for user’s preference (potential preferences) as well as user’s knowledge about the domain through interactions of information retrieval and navigation.
3. DECISION SUPPORT SYSTEM WITH SPOKEN DIALOGUE INTERFACE

3.1 System overview

The dialogue system constructed by us has two functions: answering users’ information requests and recommending them information. When the system is requested to explain about the spots or their determinants, it explains the sightseeing spots in terms of the requested determinant. This is the answering function. After providing the requested information, the system then provides information that would be helpful in making a decision (e.g., instructing what the system can explain, recommending detailed information of the current topic that the user might be interested in, etc.). This is the recommendation function. The system flow based on these strategies is summarized below. The system:

1. Recognizes the user’s query
2. Identifies the requested spot and determinant from the user’s utterance
3. Answers the query based on the result of 2
4. Recommends information related to the current topic

Note that Step 4 is optimized via reinforcement learning in Section 5.

3.2 Knowledge base

Our back-end DB consists of 15 sightseeing spots as alternatives and 10 determinants described for each spot. We select determinants that frequently appear in our corpus [Ohtake et al. 2009]. Example determinants are listed in Table IV. Normally, these determinants are related and are dependent on one another, (e.g., a user may want to see cherry blossoms and fall foliage because he/she likes nature), but in practice, the determinants are assumed to be independent and to have a parallel structure. The spots are annotated in terms of these determinants if they apply to them. The value of the evaluation $e_{nm}$ is “1” when the spot $n$ applies to the determinant $m$ and “0” when it does not.

The text is generated by retrieving appropriate reasons from the Web. An example of the DB is shown in Table I.

3.3 Spoken language understanding and response generation

Our ASR system consists of ATRASR [Itoh et al. 2004]. A trigram language model for the ASR system was trained using Wikipedia, a dialogue corpus from a different domain, and Web texts [Misu and Kawahara 2006]. A trigram language model was trained with a vocabulary of 55K words.

Our spoken language understanding (SLU) process tries to detect sightseeing spots and determinant information in the automatic speech recognition (ASR) results. We thus prepare two modules for the spots and the determinants, respectively. In order to facilitate flexible understanding, we adopt an example-based understanding method based on vector space models. That is, the ASR results are matched against a set of documents written about the target spots, and the spots

---

1 The number of alternatives is small compared to systems dealing with information retrieval, but note that this work focus on the process of comparing and evaluating candidates that meet “essential condition” (e.g. Famous temple easily accessible on foot from Kyoto station).

2 The documents we refer to are all sourced from Wikipedia.
with the highest matching scores are used as understanding results. The ASR results are also matched against a set of sample query sentences and the determinants are detected. In addition, we also concatenate contextual information on a spot or on a determinant under current focus if the ASR results included either a spot or a determinant. The system then generates a response by selecting one of the appropriate responses in the DB, and presents it through synthesized speech.

3.4 System initiative recommendation

The content of the recommendation is determined based on one of the following six methods. The dialogue act (or action) of system recommendation $a_{sys}$ consists of a communicative act $ca$ (or recommendation methods) and semantic content $sc$. The semantic content includes spots and/or determinants, which are determined by the heuristic rules defined per method.

(1) **Recommendation of determinants based on the currently focused spot (Method 1)**

This method is structured on the basis of the user’s current focus on a particular spot. Specifically, the system selects several determinants related to the current spot whose evaluation is “1” and presents them to the user. (For example, the system may recommend “Japanese garden”, “fall foliage” and “stroll information” about “Ryoanji temple”.)

(2) **Recommendation of spots based on the currently focused determinant (Method 2)**

This method functions on the basis of the focus on a certain specific determinant. The system selects several spots related to the current determinant whose evaluation is “1”. (e.g. “Kyoto Imperial Palace”, “Sanzenin” and “Kimmi’s temple” are each famous for their fall foliage.)

---

Table I. Example of the database (translation of Japanese)

| Spot name          | Determinant | Eval | Text                                                                 |
|--------------------|-------------|------|----------------------------------------------------------------------|
| Kiyo-mizu temple   | Cherry blossoms | 1    | There are about 1,000 cherry trees in the temple ground. Best of all, the vistas from the main temple are amazing. |
|                    | Vista       | 1    | The temple stage is built on a slope, and the town’s view from here is breathtaking. |
|                    | Not Crowded | 0    | This temple is highly famous and popular and thus is constantly crowded. |
| Imperial Palace    | Cherry blossom | 0    | A variety of cherry blossoms are seen in each place in the palace. In a special sighting, you can visit the cherry blossoms of “sakon”. |
|                    | Vista       | 0    | N/A                                                    |
|                    | Not Crowded | 1    | It is not crowded even though it is located at the center of Kyoto. You can enjoy your stroll as you appreciate the wonders of the palace. |

...
(3) **Open prompt (Method 3)**

The system does not make a recommendation, and presents an open prompt. After users acknowledge the domain and the system knowledge this system may be considered preferable, since users may get irritated with repetitive recommendations.

(4) **Listing of determinants 1 (Method 4)**

This method lists several determinants to the user in ascending order from the low level user knowledge $K_{sys}$. (that the system estimates). ($K_{sys}$, $P_{sys}$, $p_m$ and $Pr(p_m = 1)$ are defined and explained in Section 5.2.)

(5) **Listing of determinants 2 (Method 5)**

This method also lists the determinants, but the order is based on the user’s high preference $P_{sys}$ (that the system estimates).

(6) **Recommendation of user’s possibly preferred spot (Method 6)**

The system recommends a spot as well as the determinants that the users would be interested in based on the estimated preference $P_{sys}$. The system selects one spot $k$ with a maximum value of $\sum_{m=1}^{M} Pr(p_m = 1) \cdot e_{k,m}$. This idea is based on the collaborative filtering which is often used for recommender systems [Breese et al. 1998; Melville et al. 2002]. This method will be helpful to users if the system successfully estimates the user’s preference; however, it will be irrelevant if the system does not.

We will represent these recommendations through a dialogue act expression, $(ca_{sys} \{sc_{sys} \})$, which consists of a communicative act $ca_{sys}$ and the semantic content $sc_{sys}$. (e.g. $Method1\{(Spot_3), (Det_3, Det_4, Det_5)\}$) An example dialogue with the system is shown in Table II.

### 4. IMPLEMENTATION OF THE TRIAL SYSTEM

In order to confirm that the user’s action and decision are concerned with his/her preference and knowledge, and to collect the statistics for user model training used for dialogue simulation, we developed a trial system and conducted a user experiment. The trial system did not estimate the user’s state, and only Methods 1-3 were used to generate recommendations randomly. The number of recommended items was set to three.

#### 4.1 User experiment

We collected test data from 72 subjects who had not used our system before. Subjects were requested to use the system to determine one sightseeing spot out of 15 alternatives. No instructions or scenarios were given, except the following example utterances: “Could you please tell me the spots famous for XXX?” and “Tell me about XXX temple”. We asked the subjects not to use their own knowledge while arriving at a decision. Further, they were requested to say the phrase “I’ll go to XXX,” signifying the commitment once they had reached a decision. Only one dialogue episode was collected per subject, since the first such dialogue session would very likely alter the level of user knowledge. The average length of a dialogue before a user communicated his/her commitment was 16.3 turns with 7.0 turns being the standard deviation. We transcribed a total of 1,752 utterances and labeled their
Table II. Example dialogue with the system (translation of Japanese)

| User (U)                                | System (S)                                                                 |
|-----------------------------------------|-----------------------------------------------------------------------------|
| S1: Hello, this is the Kyoto tourist navigation system. How may I help you? | S2-Rec.: The cherry blossoms of Ninnaji are specially referred to as "omuro no sakura". These cherry trees are quite unique because they are lower than the others. |
| U1: Please tell me about the cherry blossom information of Ninnaji temple. | S2-Ans.: Japanese garden, events and fall foliage are some of the areas about which information is available on Ninnaji; do you want me to explain? (Method 1{\{Spot8\}, \{Det1, Det6, Det10\}}) |
| S2-Ans.: The cherry blossoms of Ninnaji are specially referred to as "omuro no sakura". These cherry trees are quite unique because they are lower than the others. | S2-Rec.: Japanese garden, events and fall foliage are some of the areas about which information is available on Ninnaji; do you want me to explain? (Method 1{\{Spot8\}, \{Det1, Det6, Det10\}}) |
| S2-Rec.: Japanese garden, events and fall foliage are some of the areas about which information is available on Ninnaji; do you want me to explain? (Method 1{\{Spot8\}, \{Det1, Det6, Det10\}}) | S6-Ans.: From the top of the famous gate, you can see Kyoto city spread out below. |
| S6-Rec.: Kiyomizu-temple, Fushimiinari and Kurama-temple are spots with great vistas. (Method 2{\{Spot2, Spot6, Spot8\}, \{Det4\}}) | S7-Ans.: Kiyomizu temple is built on the side of a hill and gives a great view of the Kyoto city center. |
| U5: Does it have a lovely landscape? | S7-Rec.: I can also provide you with world heritage information, stroll information and history information about the listed spots (Method 4{\{NULL\}, \{Det3, Det8, Det9\}}). |
| S6-Ans.: From the top of the famous gate, you can see Kyoto city spread out below. | U6: Kiyomizu temple. |
| S7-Rec.: I can also provide you with world heritage information, stroll information and history information about the listed spots (Method 4{\{NULL\}, \{Det3, Det8, Det9\}}). | S7-Rec.: I can also provide you with world heritage information, stroll information and history information about the listed spots (Method 4{\{NULL\}, \{Det3, Det8, Det9\}}). |
| U6: Kiyomizu temple. | S7-Rec.: I can also provide you with world heritage information, stroll information and history information about the listed spots (Method 4{\{NULL\}, \{Det3, Det8, Det9\}}). |
| S7-Rec.: I can also provide you with world heritage information, stroll information and history information about the listed spots (Method 4{\{NULL\}, \{Det3, Det8, Det9\}}). | U16: I decided to go to Nanzenji temple |
| S16: The item, Nanzenji temple, has been chosen. Are there any questions? | ...

Table III. Percentages of user utterance types for system prompts

| User Utterance Type                        | Method 1 (%) | Method 2 (%) | Method 3 (%) |
|--------------------------------------------|--------------|--------------|--------------|
| Acceptance of recommendation               | 71.5         | 30.0         | -            |
| Determinant                                | 7.0          | 25.0         | 38.6         |
| Spot name                                  | 16.8         | 16.7         | 24.8         |
| Determinant + spot name                    | 1.4          | 6.7          | 2.0          |
| Others (Commitment, OOD, etc)              | 3.3          | 21.6         | 34.6         |

4.1.1 Analysis of user utterances. First, we analyzed the relationship between system prompts and user utterances. The percentages of user utterances for system prompts (bigram probability of \(ca\)) are shown in Table III.

"Acceptance of recommendation" refers to the cases where the users accept the recommendation. That is, Method 1 is regarded as accepted when the user requests information on either of the recommended determinants. Method 2 is regarded as accepted when the user requests information on either of the recommended spots. "Determinant", "Spot name" and "Determinant + Spot name" refers to the case where the users requested information on a non-recommended determinant and/or spot. Many users make queries that the systems cannot handle (out of domain; OOD) in the open prompt (Method 3). Meanwhile, many users can make in-domain queries by presenting system knowledge through recommendations.
Table IV. Analysis of preference and knowledge

| Determinant          | Percentage of users who emphasize | Percentage of users who stated | Percentage of users stated before system recommendation |
|----------------------|----------------------------------|-------------------------------|------------------------------------------------------|
| Japanese garden      | 34.7                             | 47.2                          | 22.2                                                 |
| Not crowded          | 19.4                             | 41.7                          | 1.4                                                  |
| World heritage       | 48.6                             | 50.0                          | 2.7                                                  |
| Vista                | 48.6                             | 22.2                          | 1.4                                                  |
| Easy access          | 16.7                             | 19.4                          | 19.4                                                 |
| Fall foliage         | 37.5                             | 47.2                          | 18.1                                                 |
| Cherry flower        | 33.3                             | 51.4                          | 13.9                                                 |
| History              | 43.1                             | 31.9                          | 12.5                                                 |
| Stroll               | 45.8                             | 38.9                          | 1.4                                                  |
| Event                | 29.2                             | 36.1                          | 8.3                                                  |

4.1.2 Analysis of user preference and domain knowledge. We analyzed the sessions in terms of the preference and domain knowledge of the subjects. Table IV lists the preferences by percentage that subjects emphasize when selecting sightseeing spots. These are based on questionnaire surveys conducted after the dialogue session. (We allowed multiple selections.) Since subjects were asked to select determinants from the list of all determinants, their selections are considered to be their preferences under the entire system and user knowledge. However, when the subjects start with the dialogue sessions, some of the above preferences may be only potential preferences, because of limited nature of the users’ knowledge about the system and their own preference.

In order to analyze the knowledge of users, we analyzed the percentage of the utterances that included the determinants before the system recommendation. The result is shown in Table IV. Several determinants were seldom stated before the system made its recommendations, even if they were important for many users. For example, “World heritage site information” and “Stroll information” were seldom stated before the system’s recommendation, despite the fact that around half of the users emphasized on them. These results show that some of users’ actual preferences remained as potential preferences before the system made it explicit (At the very least, the users were not aware that the system was able to explain those determinants); thus, it is important to have users notice their potential preferences.

4.2 Analysis of users’ decisions

We then analyzed the relationship between the questionnaire-based user preference $p_{user}$ and the decided spot. We evaluated the number of agreements between the user preference and the decided spot ($\sum_{m=1}^{M} p_m \cdot c_{km}$) assuming $p_m = 1$ when the user selected the determinant in the questionnaire. The average number of agreements was 2.20, which was higher than the expectation by random selection (1.96). However, if the users had known about their potential preferences and the system knowledge, and then selected an optimal spot according to their preferences, the number ($\max_k \sum_{m=1}^{M} p_m \cdot c_{km}$) would have been 3.34.
5. OPTIMIZATION OF DIALOGUE STRATEGY

The result of the previous section revealed that an improved recommendation strategy can help users make a better choice. Thus, we optimize the dialogue strategy.

5.1 Models for simulating a user

5.1.1 User modeling. We introduce a user model that consists of a tuple of knowledge vector $K_{user}$, preference vector $P_{user}$, and local weight matrix $V_{user}$. In this paper, for simplicity, a user’s preference vector or weight for determinants $P_{user} = (p_1, p_2, \ldots, p_M)$ is assumed to consist of binary parameters. That is, if the user is interested in (or potentially interested in) the determinant $m$ and emphasizes it when making a decision, the preference $p_m$ is set to “1”. Otherwise, it is set to “0”. In order to represent a state that the user has potential preference, we introduce a knowledge parameter $K_{user} = (k_1, k_2, \ldots, k_M)$ that represents if the user has a perception that the system is able to handle or he/she is interested in the determinants. $k_m$ is set to “1” if the user knows (or listed by system’s recommendations) that the system can handle determinant $m$ and “0” when he/she does not. For example, the state that the determinant $m$ is the potential preference of a user (but he/she is unaware of that) is represented by $(k_m = 0, \ p_m = 1)$. This idea is in contrast to previous researches that assume some fixed goal that is observable by the user from the beginning of the dialogue (e.g. [Schatzmann et al. 2007]). A user’s local weight $v_{nm}$ for spot $n$ in terms of determinant $m$ is set to “1”, when the system lets the user know that the evaluation of spots are “1” through recommendation Methods 1, 2 and 3.

5.1.2 User simulator. We constructed a user simulator that is based on the statistics of Table III as well as knowledge and preference of the user. That is, the user’s communicative act $ca_{user}$ and the semantic content $sc_{user}$ for the system’s recommendation $a_{sys}$ are generated based on the following equation.

$$
Pr(ca_{user}, sc_{user}|ca_{sys}, sc_{sys}, K_{user}, P_{user}) = Pr(ca_{user}|ca_{sys}) \cdot Pr(sc_{user}|K_{user}, P_{user}, ca_{user}, ca_{sys}, sc_{sys})$$

This means that the user’s communicative act $ca_{user}$ is sampled based on the conditional probability of $Pr(ca_{user}|ca_{sys})$ in Table III. The statistic of user actions given the recommendation Method 1 is used for that for Methods 4-6. The semantic content $sc_{user}$ is selected based on the user’s preference $P_{user}$ under current knowledge about the determinants $K_{user}$. That is, the $sc$ is sampled from the determinants within the user’s knowledge ($k_m = 1$) based on the probability that the user requests the determinant of his/her preference/non-preference, which is also calculated by the dialogue data of the trial system.

---

3The user is supposed to determine the weight based on the system’s explanation about the spot and the system is also supposed to estimate that based on the user’s feedbacks.
5.2 Dialogue state expression

We define the state expression of the user in the previous section. However the problem is that for the system, the state \((P_{user}, K_{user}, V_{user})\) is not observable, but is only estimated from the interactions with the user. Thus, this model is a partially observable Markov decision process (POMDP) problem. In order to estimate unobservable properties of POMDP and handle the problem as MDP, we introduce the system’s inferential user knowledge vector \(K_{sys}\) or probability distribution (estimate value) \(K_{sys} = \left( Pr(k_1 = 1), Pr(k_2 = 1), \ldots, Pr(k_M = 1) \right)\) and that of preference \(P_{sys} = \left( Pr(p_1 = 1), Pr(p_2 = 1), \ldots, Pr(p_M = 1) \right)\). The estimated local weight \(V_{sys}\) is supposed to be introduced. This idea is similar to the belief state approach [Thrun 2000] used in [Williams and Young 2007]. In this work, we do not estimate the weight, because \(v_{nm}\) is assumed to be set to “1” only when the system lets the user know that the evaluation of the determinant \(m\) of the spot \(n\) is “1” through recommendations, thus \(V_{sys} = V_{user}\). This consequently means that \(V_{user}\) is observable.

The dialogue state \(DS^{t+1}\) or estimated user’s dialogue state of the step \(t + 1\) is assumed to be dependent only on the previous state \(DS^t\), as well as the interactions \(I^t = (a_{sys}^t, a_{user}^t)\). This approximation is often adopted in many spoken dialogue management system as dynamic Bayesian network (DBN) expression [Pietquin and Dutoit 2006; Thomson et al. 2008; Minami et al. 2009]. The relation of the parameters used in our model is illustrated as DBN in Figure 2.

The estimated user’s state is represented as a probability distribution and is updated by each interaction. This corresponds to representing the user types as a probability distribution, whereas the work of [Komatani et al. 2005; Paksima et al. 2009] classifies users to several discrete user types. The estimated user’s preference \(P_{sys}\) is updated when the system observes the interaction \(I^t\). The update is conducted based on the following Bayes’ theorem using the previous state \(DS^t\).
Priors of the estimated state:
- Knowledge:
  \( K_{sys} = (0.22, 0.01, 0.02, 0.18, \ldots) \)
- Preference:
  \( P_{sys} = (0.37, 0.19, 0.48, 0.38, \ldots) \)

Interactions (observation):
- System recommendation:
  \( a_{sys} = \text{Method}1\{(\text{Spot}_2, (\text{Det}_1, \text{Det}_3, \text{Det}_4)) \}
  \)
  “Japanese garden (\text{Det}_1), World heritage (\text{Det}_3) and fall foliage (\text{Det}_4) are some of the areas about which information is available on Ninnaji (\text{Spot}_2).”
- User query:
  \( a_{user} = \text{Accept}\{(\text{Spot}_2), (\text{Det}_3)\}
  \)
  “Tell me about the World heritage. (\text{Det}_3)”

Posterior of the estimated state:
- Knowledge:
  \( K_{sys} = (1.00, 0.01, 1.00, 1.00, \ldots) \)
- Preference:
  \( P_{sys} = (0.26, 0.19, 0.65, 0.22, \ldots) \)

User’s knowledge acquisition:
- Knowledge:
  \( K_{user} \leftarrow \{k_1 = 1, k_3 = 1, k_4 = 1\} \)
- Local weight:
  \( V_{user} \leftarrow \{v_{51} = 1, v_{53} = 1, v_{54} = 1\} \)

Fig. 3. Example of state update

as a prior. Although it is possible to explicitly ask the user if he/she is interested in the determinants, we consider an implicit preference estimation. Further, \( p_m \) would not necessarily be “1” even if the user positively answered such a question.

\[
Pr(p_m = 1|I^t) = \frac{Pr(I^t|p_m = 1)Pr(p_m = 1)}{Pr(I^t|p_m = 1)Pr(p_m = 1) + Pr(I^t|(p_m = 0))Pr(1 - Pr(p_m = 1))}
\]

Here, \( Pr(I^t|p_m = 1), Pr(I^t|(p_m = 0)) \) to the right side was obtained from the dialogue corpus of Section 4. (e.g., If the user requests information about some determinant \( k \) immediately after the system’s open prompt (Method 3), his/her preference of the determinant \( p_k \) is more likely to be “1” than the case that he/she requests after recommendation by Method 1. The posterior of the estimated user’s knowledge of determinants \( k_m \) is updated to “1” when the system tells or the user requests the determinants. This posterior is then used as a prior in the next state update using interaction \( I^{t+1} \). An example of this update is illustrated in Figure 3.

5.3 Reward function

The reward function that we use is based on the number of agreed attributes between the user preference and the decided spot.

Users are assumed to determine the spot based on their preference \( P_{user} \) under their knowledge \( K_{user} \) (and local weight for spots \( V_{user} \)) at that time, and select the spot \( k \) with the maximum priority of \( \sum_k k \cdot p_k \cdot v_{km} \).

The reward \( R \) is then calculated based on the improvement in the number of
agreed attributes between the user’s actual (potential) preferences and the decided spot \( k \) over the expected agreement by random spot selection.

\[
R = \sum_{m=1}^{M} p_{m} \cdot e_{k,m} - \frac{1}{N} \sum_{n=1}^{N} \sum_{m=1}^{M} p_{m} \cdot e_{n,m}
\]

For example, if the decided spot satisfies three preferences and the average agreement of the agreement by random selection is 1.3, then the reward is 1.7. If more than two spots have the optimal priority for decision, the average of the reward by those spots is given.

5.4 Optimization by reinforcement learning

5.4.1 Definition of MDP. The problem of system recommendation generation is optimized through reinforcement learning. The MDP \((S, A, R)\) is defined as the following factors. The state parameter \( S = (s_1, s_2, \ldots, s_I) \) (that corresponds to feature vector of the current dialogue state) is generated by extracting the features of the current dialogue state \( DS_t \). We use the following 29 features

1. Parameters that indicate the \# of interactions from the beginning of the dialogue. This is approximated by five parameters using triangular functions.
2. User’s previous action (communicative act) (1 if \( a_{user}^{t-1} = x_i \), otherwise 0)
3. System’s previous action (communicative act) (1 if \( a_{sys}^{t-1} = y_j \), otherwise 0)
4. Sum of the estimated user knowledge about determinants (\( \sum_{n=1}^{N} Pr(k_n = 1) \))
5. Number of presented spot information. (This consequently corresponds to \( \sum_{n=1}^{N} \sum_{m=1}^{M} v_{nm} \) in this work)
6. Expectation of the probability that the user emphasizes the determinant in the current state (\( Pr(k_n = 1) \times Pr(p_n = 1) \)) (10 parameters)

The action set \( A \) consists of the six recommendation methods shown in subsection 3.4. Reward \( R \) is given by the reward function of subsection 5.3.

5.4.2 Policy for system action generation. A system action \( a_{sys} (ca_{sys}) \) is sampled based on the following soft-max (Boltzmann) policy.

\[
\pi(a_{sys} = k|S) = Pr(a_{sys} = k|S, \Theta) = \frac{\exp(\sum_{i=1}^{I} s_i \cdot \theta_{ki})}{\sum_{j=1}^{J} \exp(\sum_{i=1}^{I} s_i \cdot \theta_{ji})}
\]

Here, \( \Theta = (\theta_{11}, \theta_{12}, \ldots, \theta_{1J}, \ldots, \theta_{IJ}) \) consists of \( J \) (# actions) \times \( I \) (# features) parameters. The parameter \( \theta_{ji} \) works as a weight for the \( i \)-th feature of the action \( j \) and determines the likelihood that the action \( j \) is selected. This \( \Theta \) is the targets of optimization by reinforcement learning.

\(^4\)

Note that about half of them are continuous valuable and that the value function cannot be denoted by a lookup table

ACM Transactions on Computational Logic, Vol. 2, No. 3, 09 2001.
5.4.3 Optimization by Natural Actor Critic. We adopt Natural Actor Critic (NAC) [Peters and Schaal 2008; Thomson et al. 2008], which adopts a natural policy gradient method as the policy optimization method. Policy gradient methods do not directly update the value of state $S$ or Q-function $Q(S,A)$. Instead, the policy $\pi$ (or parameter $\Theta$ in our case) is directly updated so as to increase the reward of dialogue episodes generated by the previous policy.

5.5 Experiment by dialogue simulation

For each session, a simulated user $(P_{user}, K_{user}, V_{user})$ is sampled based on the statistics of the trial system 4. A preference vector $P_{user}$ of the user is generated so that he/she has four preferences. As a result, four parameters in $P_{user}$ is “1” and the others are “0”. This vector is fixed throughout the dialogue episode. This sampling is conducted based on the rate proportional to the percentage of users who emphasize it for making decisions (c.f. Table IV). The user’s knowledge $K_{user}$ is also set based on the statistics of the “percentage of users who stated the determinants before system recommendation”. For each determinant, we sample a random valuable $r$ that ranges from “0” to “1”, and $k_m$ is set to “1” if $r$ is smaller than the percentage. All the parameters of local weights $V_{user}$ are initialized to “0”, assuming that users have no prior knowledge about the candidate spots. As for system parameters, the estimated user’s preference $P_{sys}$ and knowledge $K_{sys}$ are initialized to the value of Table IV and used as a prior.

We first assumed that the system does not misunderstand the user’s action. Users are assumed to continue a dialogue session for 20 turns, and episodes are samples using the policy $\pi$ at that time and the user simulator of subsubsection 5.1.2. The system is rewarded by the reward function of subsection 5.3 and the policy (parameter $\Theta$) is updated using NAC in every 2,000 dialogues.

5.6 Experimental result

Before discussing the optimization, we discuss the necessity of recommendations in spoken decision-making dialogue under restricted knowledge. We compared our user model with the model with users’ entire knowledge about the determinants that system can handle and his/her interests lie, and use the system in order to find the spot that applies to the interest (= to fix local weight matrix $V_{user}$). We conducted a simulation using a strategy with no recommendation (This is equivalent to the strategy that always chooses the Method 3). Table V lists comparison of the average reward between our user model and that with entire knowledge after 10, and 20 turns of interaction. The reward by our user model is much smaller than that of the model assuming that the user has entire knowledge. This result suggests that the system has to bridge gaps of knowledge in such a spoken decision-making dialogue.

5.6.1 Improvement by policy iteration. We first examined the improvement by policy iterations. Since the method has a random factor in the action generation, the result is an average of five trials. Figure 4 illustrates the relationship between

---

Footnote: In practice, users may make a decision at any point once he/she satisfied collecting information. And this is the reason why we list the rewards in the early dialogue stage in the following subsections.
Table V. Comparison of reward with user model with entire knowledge on determinant ans his/her preference

| Policy                  | Reward (±std) |       |       |
|-------------------------|---------------|-------|-------|
|                         | T = 10        | T = 20|
| Our user model          | 0.13 (0.54)   | 0.34 (0.59)|
| With entire knowledge   | 0.45 (0.60)   | 0.70 (0.62)|

![Graph showing reward by step (turn) T](image)

Fig. 4. # batches v.s. Reward by step (turn) T

the number of batches (2,000 dialogue episodes) and average reward after 2, 5, 10, 15 and 20 turns of interaction. The reward by optimal decision under the condition that the user has entire knowledge about the knowledge base (including $V_{user}$)\(^6\), which is the upper limit of the reward, was also illustrated (Oracle). The policy was fixed about 30,000 dialogue episodes.

We analyzed the learned dialogue policy by examining the value of weight parameter $\Theta$. We compared the parameters of the trained policy between actions\(^7\). The weight of the parameters that represent the early stage of the dialogue was large in Methods 4 and 5. On the other hand, the weight of the parameters that represent the latter stage of the dialogue was large in Methods 2 and 6. This suggests that in the trained policy, the system first bridges knowledge gap between the user, estimates the user’s preference, and then, recommends specific information that would be useful to the user. This flow is similar to the strategy of human guide collected in our dialogue corpus [Ohtake et al. 2009].

5.6.2 Evaluation of learned policy. Next, using the trained weight parameter, another simulation, where the system selected the recommendation method using the greedy policy that always selects best action (instead of soft-max policy), was conducted. We compared the trained policy with the following baseline methods.

(1) No recommendation (B1)

The system only provides the requested information and does not generate any recommendations.

\(^6\)At least 50 turns are required for this.

\(^7\)The parameters can be interpreted as the size of the contribution for selecting the action.
The system randomly chooses a recommendation from six methods. This corresponds to the policy by the initial parameter of $\Theta$.

The comparison of the average reward between the baseline methods is listed in Table VI. The reward by the strategy optimized by NAC was significantly better than that of baseline methods ($n=500, p<.01$).

5.6.3 Evaluation of features about estimated user’s knowledge and preference.
We then compared the proposed method with the case where estimated user’s knowledge and preference is represented as a discrete binary parameters instead of probability distributions (PDs). That is, the estimated user’s preference $p_m$ of determinant $m$ is set to “1” when the user requested the determinant, otherwise it is “0”. The estimated user’s knowledge $k_m$ is set to “1” when the system lets the user know the determinant, otherwise it is “0”. Another dialogue strategy was trained using this dialogue state expression. This result is shown in Table VII. The proposed method that represents the dialogue state as a probability distribution outperformed ($p<.01$ (T=15,20)) the method using a discrete state expression.

We also compared the proposed method with the case where either one of estimated preference or knowledge was used for feature for dialogue state in order to carefully investigate the effect of these factors. In the proposed method, expectation of the probability that the user emphasizes the determinant ($Pr(k_n=1) \times Pr(p_n=1)$) was used as a feature of dialogue state. We evaluated the performance of the cases where estimated knowledge $Pr(k_n=1)$ or estimated preference $Pr(p_n=1)$ was used instead of the expectation of the probability that the user emphasizes the determinant. This result is shown in Table VIII. We confirmed that significant im-
Table IX. Comparison of reward with baseline methods (with penalty for recommendations)

| Policy | T = 5   | T = 10  | T = 15  | T = 20  |
|--------|---------|---------|---------|---------|
| NAC    | 0.94 (0.52) | 0.93 (0.51) | 0.93 (0.50) | 0.95 (0.51) |
| B1     | 0.02 (0.42)  | 0.13 (0.54)  | 0.29 (0.59)  | 0.34 (0.59)  |
| B2     | 0.30 (0.67)  | 0.31 (0.66)  | 0.25 (0.62)  | 0.14 (0.58)  |

Table X. Evaluation considering errors in ASR

| Policy | T = 5   | T = 10  | T = 15  | T = 20  |
|--------|---------|---------|---------|---------|
| NAC    | 0.95 (0.55) | 1.01 (0.55) | 1.09 (0.55) | 1.14 (0.52) |
| B1     | 0.02 (0.41)  | 0.13 (0.55)  | 0.26 (0.58)  | 0.31 (0.58)  |
| B2     | 0.42 (0.68)  | 0.65 (0.66)  | 0.79 (0.63)  | 0.89 (0.58)  |

provement ($p < .01$ (T=15,20)) was obtained by taking into account the estimated knowledge of the user.

5.6.4 Evaluation considering penalty by recommendations. We discuss the problem of trade-off between the length of system prompt and the improvement of the decision. Information recommendations may irritate the user, especially when he/she clearly knows what to ask the next. Repetitive recommendations may also irritate the user. We thus evaluated the performance by penalizing recommendation. (The balance is supposed to be estimated by factor analysis methods, e.g., PARADISE [Walker et al. 1997].) We assumed that each system recommendation except the open prompt (Method 3) is penalized by 0.05. Another dialogue strategy was trained using this reward function. This result is shown in Table IX. The trained strategy was less degraded as compared to the random strategy when the dialogue was protracted. Thus, it is considered that the strategy was trained so as to avoid unnecessary recommendation. These rewards in all points were statistically significant ($p < .01$) over the baseline methods. These figures demonstrate the effectiveness of the trained dialogue strategy.

5.6.5 Evaluation considering ASR errors. Finally, we evaluated the performance, considering errors in ASR [Schatzmann et al. 2007; Pietquin et al. 2009]. Since our language model adopt a statistical language model with a large lexicon, most ASR errors are deletion errors. We thus assumed that the semantic content is deleted in the percentage of subsection 4.1. The communicative act of the user utterance was also replaced according to the change of the semantic contents in the utterance.

This result is shown in Table X. The reward by the proposed state expressions was less degraded compared to the performance of ASR (83.3% of original performance). In addition, the rewards were still better than that of baseline methods.

6. CONCLUSION

In this paper, we addressed a spoken dialogue framework that helps users select an alternative from a list of alternatives. Specifically, we proposed a user model and a manner of dialogue state expression in spoken decision making dialogue. User knowledge as well as preference are considered in the models and the system dynamically update the estimated user state through interactions. The dialogue
strategy of the system was optimized by reinforcement learning using a user simulator trained by large number of dialogue data collected using a trial dialogue system. We confirmed that the learned policy achieved a better recommendation strategy over several baseline methods.

One important assumption in the work is the number of candidates. We assumed that the number of candidates are relatively small focusing on the process of comparing and evaluating candidates that meet “essential conditions”. However, several interactions would be required before arranging such a situation, thus the optimization of dialogue strategy considering such a process to narrow down candidates is our future work.

Another important assumption lies in the parameters of user model. In the work, a user’s preference vector or weight for determinants $P_{user}$ and $V_{user}$ is assumed to consist of binary parameters. However, in practice, the parameters may take any positive numbers (= Users do not equally emphasize on the determinants they are interested in and users’ evaluations about spots in terms of determinants will vary with spot.), and without correctly estimating the values, it would be difficult to properly evaluate real users’ decision. We believe this could be solved by estimating these parameter as regression problem.

Although we dealt with a simple recommendation strategy, there are many possible extensions to this model. The system is expected to handle a more complex planning of natural language generation, such as those discussed in [Rieser and Lemon 2009]. We can also extend the model to estimate the local weights $V_{user}$ from implicit feedbacks of the user (e.g., backchannel and utterance timing [Kawahara et al. 2008]). We also need to consider more complex errors simulation and error handling in speech recognition and understanding when simulating dialogue [Schatzmann et al. 2007; Pietquin et al. 2009].

Despite these limitations, the result of the experiments by the proposed methods here are promising. We would like to extend the work to overcome the limitations.

REFERENCES

BOHUS, D., LANGNER, B., RAUX, A., BLACK, A., ESKENAZI, M., AND RUDNICKY, A. 2006. Online Supervised Learning of Non-understanding Recovery Policies. In Proc. Spoken Language Technology Workshop (SLT). 170–173.

BRESEE, J., HECHEMANN, D., AND KADIE, C. 1998. "empirical analysis of predictive algorithms for collaborative filtering". In "Proc. the 14th Annual Conference on Uncertainty in Artificial Intelligence". 43–52.

DOHSAKA, K., YASUDA, N., AND AIKAWA, K. 2003. Efficient Spoken Dialogue Control Depending on the Speech Recognition Rate and System’s Database. In Proc. Eurospeech.

ITOH, G., ASHIKARI, Y., JITSUHIRO, T., AND NAKAMURA, S. 2004. Summary and evaluation of speech recognition integrated environment ATRASR. In Proc. 2005 Acoustic Society of Japan Fall Meeting [in Japanese]. 221–222.

KAWAHARA, T., TOYOKURA, M., MISU, T., AND HORI, C. 2008. Detection of Feeling Through Back-Channels in Spoken Dialogue. In Proc. Interspeech. 1696–1696.

KOMATANI, K., UENO, S., KAWAHARA, T., AND OKUNO, H. 2005. User Modeling in Spoken Dialogue Systems to Generate Flexible Guidance. User Modeling and User-Adapted Interaction 15, 1, 169–183.

LAMEL, L., BENNACEF, S., GAUVAIN, J. L., DARTIGUES, H., AND TEMMEM, J. N. 2002. User Evaluation of the MASK Kiosk. Speech Communication 38, 1.
Lamel, L., Rosset, S., Gauvain, J., and Bennacef, S. 1999. The LIMSI ARISE System for Train Travel Information. In Proc. ICASSP.

Levin, E., Pieraccini, R., and Eckert, W. 2000. A Stochastic Model of Human-machine Interaction for Learning Dialogue Strategies. IEEE Trans. on Speech and Audio Processing 8, 11–23.

Marchionini, G. 2006. Exploratory search: from finding to understanding. Communications of the ACM 49, 4, 41–46.

Melville, P., Mooney, R., and Nagarajan, R. 2002. Content-Boosted Collaborative Filtering for Improved Recommendations. In Proc. of Eighteenth National Conference on Artificial Intelligence. 187–192.

Minami, Y., Mori, A., Meguro, T., Higashinaka, R., Dohsaka, K., and Maeda, E. 2009. Dialogue Control Algorithm for Ambient Intelligence based on Partially Observable Markov Decision Processes. In Proc. First International Workshop on Spoken Dialogue Systems Technology (IWSDS).

Misu, T. and Kawahara, T. 2006. A Bootstrapping Approach for Developing Language Model of New Spoken Dialogue Systems by Selecting Web Texts. In Proc. Interspeech.

Ohtake, K., Misu, T., Hori, C., Kashoka, H., and Nakamura, S. 2009. Annotating Communicative Function and Semantic Content in Dialogue Act for Construction of Consulting Dialogue Systems. In Proc. Interspeech.

Paksima, T., Georgila, K., and Moore, J. 2009. Evaluating the Effectiveness of Information Presentation in a Full End-To-End Dialogue System. In Proc. SIGDIAL. 1–10.

Peters, J. and Schaal, S. 2008. Natural Actor-Critic. Neurocomputing 71, 7-9, 1180–1190.

Pietquin, O. and Dutoit, T. 2006. A probabilistic framework for dialog simulation and optimal strategy learning. IEEE Trans. on Audio, Speech and Language Processing 14, 2, 589–599.

Pietquin, O., Rossignol, S., and Ianotto, M. 2009. Training Bayesian networks for realistic man-machine spoken dialogue simulation. In Proc. First International Workshop on Spoken Dialogue Systems Technology (IWSDS).

Pinault, F., Lefevre, F., and de Mori, R. 2009. Feature-based Summary Space for Stochastic Dialogue Modeling with Hierarchical Semantic Frames. In Proc. Interspeech.

Polifroni, J. and Walker, M. 2008. Intensional Summaries as Cooperative Responses in Dialogue: Automation and Evaluation. In Proc. ACL/HLT. 479–487.

Potamianos, A., Ambicht, E., and Kuo, H. 2000. Dialogue Management in the Bell Labs Communicator System. In Proc. ICSLP.

Raux, A., Langner, B., Black, A., and Eskenazi, M. 2005. Let’s Go Public! Taking a Spoken Dialog System to the Real World. In Proc. Interspeech.

Rieser, V. and Lemon, O. 2009. Natural Language Generation as Planning Under Uncertainty for Spoken Dialogue Systems. In Proc. 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL).

Rudmicky, A., Bennett, C., Black, A., Chotomongcol, A., Lenzo, K., Oh, A., and Singh. 2000. Tasks and Domain Specific Modelling in the Carnegie Mellon Communicator System. In Proc. ICSLP. Vol. 2.

Saaty, T. 1980. The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation. Mcgraw-Hill.

Schatzmann, J., Thomson, B., Weilhammer, K., Ye, H., and Young, S. 2007. Agenda-based User Simulation for Bootstrapping a POMDP Dialogue System. In Proc. HLT/NAACL.

Schatzmann, J., Thomson, B., and Young, S. 2007. Error simulation for training statistical dialogue systems. In Proc. Automatic Speech Recognition and Understanding Workshop (ASRU). 9–13.

ACM Transactions on Computational Logic, Vol. 2, No. 3, 09 2001.
Seneff, S. and Polifroni, J. 2000. Dialogue management in the Mercury flight reservation system. In Proc. ANLP-NAACL 2000, Satellite Workshop.

Sturm, J., Os, E., and Boves, L. 1999. Issues in spoken dialogue systems: Experiences with the Dutch ARISE system. In Proc. ESCA workshop on Interactive Dialogue in Multi-Modal Systems.

Thomson, B., Schatzmann, J., and Young, S. 2008. Bayesian Update of Dialogue State for Robust Dialogue Systems. In Proc. ICASSP. 4937–4940.

Thrun, S. 2000. Monte carlo POMDPs. In Advances in Neural Information Processing Systems 12. 1064–1070.

Walker, M., Litsman, D. J., Kamm, C., and Abella, A. 1997. PARADISE: A Framework for Evaluating Spoken Dialogue Agents. In Proc. of the 35th Annual Meeting of the ACL.

Williams, J. and Young, S. 2007. Scaling POMDPs for Spoken Dialog Management. IEEE Trans. on Speech and Audio Processing 15, 7, 2116–2129.

Zue, V., Seneff, S., Glass, J., Polifroni, J., Pao, C., Hazen, T., and Hetherington, L. 2000. Jupiter: A Telephone-based Conversational Interface for Weather Information. IEEE Trans. on Speech and Audio Processing 8, 1, 85–96.