Guiding the User when Searching Information on the Web

Marta Gatius **Meritxell González**
Technical University of Catalonia
Barcelona, Spain
{gatius,mgonzalez}@lsi.upc.edu

Abstract

This paper describes how we approach the problem of guiding the user when accessing informational web services. We developed a mixed-initiative dialogue system that provides access to web services in several languages. In order to facilitate the adaptation of the system to new informational web services dialogue and task management were separated and general descriptions of the several tasks involved in the communication process were incorporated.

1Introduction

This paper describes how we approach the problem of guiding the user when accessing informational web services. We designed a dialogue system (DS) for accessing different types of applications in several languages. The results of the evaluation of the first prototype are described in (Gatius and González, 2009). In order to improve both the functionality and adaptability of the DS we have studied the most appropriate representation of the general and application-specific conceptual knowledge involved when helping the user to access informational services.

When providing access to information-seeking applications DSs use an underspecified set of constraints to restrict the search rather than a defined user's goal (which can be broken down into tasks and subtasks). Hence, the main tasks for DSs providing access to an informational service consist of guiding the user to give the needed constraints as well as presenting in an appropriate way the results. There have been several approaches to face this problem (Rieser and Lemon, 2009; Steedman and Petrick, 2007; Varges et al., 2009). Our approach consists of separating completely dialogue management from task management (following other relevant proposals (Allen et al., 2001)), and defining the general tasks involved when accessing informational services. Besides, general mechanisms using the two main knowledge bases of the system (the dialogue context and the domain conceptual knowledge) are used to relax the query constraints and to state additional constraints.

2 Dialogue and Task Management

The DS we developed consists of five independent modules: the language understanding, the dialogue manager (DM), the task manager, the language generator and the user model, used to adapt automatically the dialogue strategies. Additionally, there are two main data structures accessible for all modules: the information state, representing the dialogue context and the conceptual knowledge, describing the application domain.

The DM follows the information state update model, which provides a complete separation of dialogue and task management. The DM uses communication plans to determine the next system actions that could satisfy user's requirements. These plans are generated (semi)-automatically when a new service is incorporated into the DS by adapting the general communication plan for the service type to the particular service specifications.

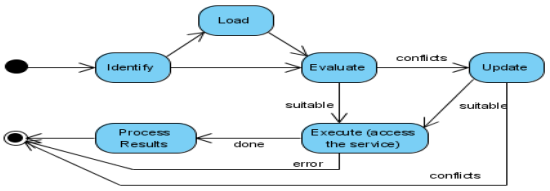

Figure 1: Task Management in the Dialogue System

Figure 1 shows task management in the DS. Main tasks performed by the task manager are the following: identification of the required web service and the specific service task, completion of the data obtained from the user, access to the service and presentation of the results.

Once the communication starts and the first intervention of the user has been interpreted and passed to the task manager, it has to identify the service and the specific service task that has to be accessed. Then, an instantiation of the specific task is generated. There are several general task descriptions for each service type, for the informational services two tasks are considered: find a list of items and describing an item.

Finally, the task manager accesses the web service and decides the most appropriate presentation of the results obtained.

3 Accessing InformationalServices

The tasks involved when the system guides the user to access an informational service are shown in Figure 2. Circled blocks represent the specific information the DM has to obtain from the user: the searched data (requestedData or output parameters) and the data constraining the query (queryConstraints or input parameters). Rectangles represent the three different tasks processing the resulting data: describing a particular item, collecting a list of items and summarizing the results obtained. Colored blocks correspond to the three different processes considered when updating constraints: relaxation, using default values and adding new constraints.

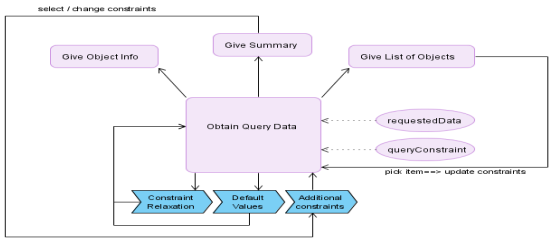

Figure 2: The tasks involved in information-seeking

The process of obtaining the query constraints from the user could be complex, as they are not gathered in a predetermined order. The task manager determines whether a complete query can be generated or if additional information has to be obtained from the user. If the service's definition includes default values, they can be included to complete the query. Parameter values appearing in previous turns can also be used.

The information obtained from the service has to be processed. Four different situations are distinguished: the result is only one item, the number of items obtained belongs to a predefined range, there are too many results and there are no results. In case there is only one item a detailed description of this item is given. In case the number of results is acceptable, a list enumerating all of them is presented to the user, suggesting him to pick up one. In the two latter cases the constraints have to be updated.

In the specific case that there are no results, the task manager can automatically relax the constraints and execute the query again. The constraints can be relaxed at the level of the query and at the level of the parameter's values. In the former, the system removes one or more of the query constraints. In the latter, the system updates the value for one or more of the constraints. The conceptual knowledge base is used to relax the constraints. If taxonomies describing the domain have been incorporated, a class is substituted by the upper class (for example, if the user asks for *drama movies* and there are none, the upper class *movies* would be used). Several strategies for data common to several applications (such as dates and locations) are already considered.

In the specific case that too many items are obtained from the service, the system presents a summary of the results. Information suggesting possible additional constraint values could also be given to the use.

5 Conclusions and Future Work

In order to improve the functionality and adaptability of our DS when guiding the user to accessing informational service we have studied the general and the application specific conceptual knowledge involved in the communication process. In our system this knowledge has been represented as a general scheme from which the communication plans for each informational service are generated and general task that are instantiated for each service. The resulting architecture facilitates the integration of other application types into the system since the task models can be easily extended and adapted.

Future work could include the processing of user's questions which answer involves the processing of data obtained from several web services.

References

J.F. Allen, G. Ferguson and A. Stent 2001. An Architecture for More Realistic Conversational Systems. Intelligent User Interfaces Conference.

M. Gatius and M. González 2009. A Flexible Dialogue System for Enhancing Web Usability. 18th World Wide Web Conference.

M. Steedman and R. Petrick 2007. Planning Dialog Actions. 8th SIGdial Workshop on Discourse and Dialogue.

V. Rieser and O. Lemon 2009. Does this list contain what you were searching for? Learning adaptive dialogue strategies for Interactive Question Answering. Natural Language Engineering, 15(1):57- 375. Cambridge University Press.

S. Varges, F. Weng and H. Ponbarry 2009. Interactive question answering and constraint relaxation in spoken dialogue systems. Natural Language Engineering,15(1).Cambridge University Press.