Service Navigation Technology Based on Task Knowledge

Takefumi Naganuma† and Shoji Kurakake

Abstract

With mobile Internet services proliferating rapidly, ordinary mobile users are finding it virtually impossible to select the most appropriate ones. We propose a system that assists non-expert mobile users to find appropriate services that solve their real-world problems. This article describes a study of a service navigation system using a knowledge base that stores user task models extracted from service-usage cases to simplify the discovery of services applicable to real-world problem solving.

1. Introduction

Mobile Internet technology is making rapid progress, especially in the field of mobile terminals. In response to this high penetration rate, the volume of Web content is ever increasing, and users can now access a wide variety of information services from mobile terminals anytime and anywhere. This should make it possible to construct an environment that can solve many of the problems that a user faces in daily life regardless of his or her location.

To provide optimal information services to users, we must develop an intelligent service-provision infrastructure for selecting and presenting optimal information services taking into account user circumstances and objectives. However, the current service-provision infrastructure is dominated by directory-based search menus and keyword-based full-text search engines. When using such a search system, a user selects some information services as candidates and then evaluates each of them using hints from index descriptions and the like. In general, an information service is used to solve a problem of some kind. This problem-solving process can be defined using a six-step process model called the “Big6 model” [1], as shown in **Fig. 1**. In this model, the support range of a conventional search engine corresponds to the third step (location and access). However, users should be able to decide the necessary information in the second step (information seeking strategies): it is not very useful if a user who has no information searching skills is required to select an appropriate information service.

Against this background, we propose a service navigation system that aims to enable users, including those with no information-searching skills, to discover information services for problem-solving purposes using a mobile terminal. For this system, we construct a knowledge base that stores user task models extracted from service-usage cases and assign the correspondence between user tasks and information services. In this article, we present a technique for constructing such a knowledge base and describe a query matching system. We also describe a prototype system that we constructed and present the results of a subjective evaluation.

† NTT DoCoMo Inc.
Yokosuka-shi, 239-8536 Japan
E-mail: naganuma@netlab.nttdocomo.co.jp

2. Service navigation system

2.1 Our approach

A general approach to problem-solving is to divide a large or abstract problem into some smaller sub-problems. For example, human daily life is driven by “proximal goals” (short-term goals), which are derived from “distal goals” (long-term goals) [2]. In the area of using information services to solve a problem, the problems that can be solved directly by using an information service correspond to proximal goals. A distal goal is achieved by solving sub-problems corresponding to proximal goals derived from the

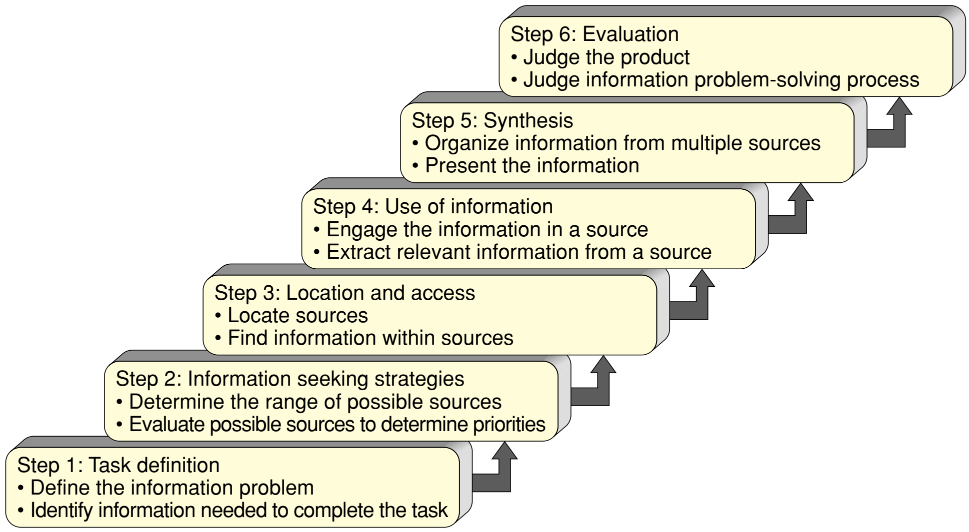

Fig. 1. Information problem-solving process (Big6 model).

distal goal [3]. Our idea is to divide the user's distal goal into a few proximal goals and guide the user to appropriate services that can solve them.

The following explains this concept using a specific example. Consider the real world problem: "I'd like to spend this weekend at an amusement park. What kind of information service is available?" The information-seeking action generated by this problem may take the following form:

Goal: go to amusement park

- decide one's schedule
- decide on ways to get to the destination
- decide on a route to the destination

In this scenario, "go to an amusement park" corresponds to the long-term goal. The three sub-problems that must be solved to achieve the long-term goal correspond to short-term goals.

To make the above approach possible, we treat "what a user wants to do" as the user's task and structure the knowledge that can be used to divide the task into sub-tasks as task knowledge. We have built a knowledge base that stores such structured task knowledge. Furthermore, we have developed knowledge about service usage (e.g., provides functions for reserving movie tickets) as service knowledge and have built a knowledge base that stores service knowledge.

2.2 System architecture

The architecture of the proposed service navigation system is shown in **Fig. 2**. This system consists of a task knowledge base that stores task knowledge; a service knowledge base that associates the task knowledge base with the uniform resource identifiers (URIs) of information services; a knowledge server that searches for appropriate task-nodes in the task and service knowledge bases and returns a reply in response to the user's query from a mobile terminal; and a client application that provides a user interface, communicates with the knowledge server, and acquires and displays a task unit, which is a set of related task nodes in the task knowledge base including a root task and its sub-tasks.

In this system, the user issues a task unit request to the knowledge server via the client application. The knowledge server analyzes the request and selects a task unit that matches that request by searching the task knowledge base. At this time, the knowledge server also searches the service knowledge base, selects the URIs of all information services associated with the selected task unit, and returns them with the task unit to the client application. The user now searches the task unit displayed on the client application and selects tasks to be executed. Then he or she selects an information service from those associated with the selected tasks.

2.3 Task knowledge base

Task knowledge has a hierarchical structure consisting of a root task node corresponding to a real-world problem perceived by the user and sub-tasks resulting from the division of that task node. Functions provided by information services (service functions) correspond to directly solvable task nodes among those included in this hierarchical structure. Therefore, it is necessary to analyze information ser-

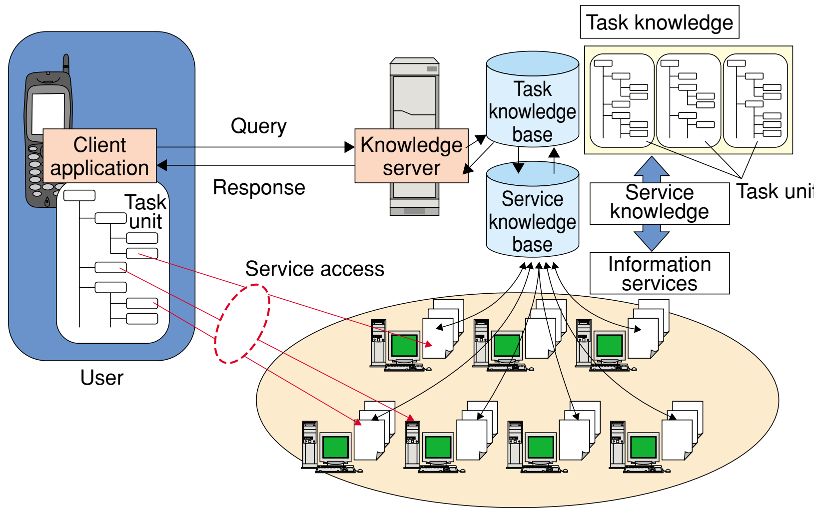

Fig. 2. Architecture of service navigation system.

vices that are being provided and extract service functions. It is then necessary to analyze specific cases (scenarios) up to the stage of using information services with the aim of integrating previously extracted service functions and structuring task knowledge.

When setting out to construct a knowledge base, one must clarify the problem areas to be targeted by the system. For this purpose, we introduced a domain model based on categories of real-world locations (e.g., amusement parks, department stores, and hotels). This model considers that one domain consists of the following three areas of activity: 1) preparation to reach the target destination, 2) travel to the target destination, and 3) behavior at the target destination. Task knowledge relates to these activity areas within a domain.

Task nodes that make up task knowledge can be represented by a generic process consisting of a noun (generic noun) that acts as the object of an action and a verb (generic verb) that indicates an action performed on an object. A problem that occurs when representing task nodes in this way is the wide range of different descriptions that result from the ambiguity of natural language. A simple example is car and automobile, which are variations of the same thing. This can lead to multiple definitions of synonymous task nodes. To solve this problem, we used a 220,000-word thesaurus to provide semantic definitions for allowable vocabulary.

(1) Extracting service functions

To define task nodes that can be directly solved through the use of information services, we must collect and analyze actual services and extract service functions. Service functions are generalizations of functions that provide Web content without losing the meaning of those original functions. They are described by a generic process just like task nodes. First, we extract Web content for mobile terminals at random from various information sources. In this sampling, we exclude irrelevant content, such as ring tones, screen wallpaper, and games, that is not applicable to user-behavior support. Next, we assign service functions to sampled Web content. For content having multiple functions, we assign multiple service functions. Finally, we merge service functions assigned to each service and integrate semantically equivalent service functions. Applying the above procedure has so far produced definitions for about 2000 service functions with respect to about 2700 items of Web content.

(2) Structuring the task knowledge based on cases

Using specific cases of behavior up to the stage of the actual use of information services, we extract upper-level task nodes and structure the task knowledge by integrating previously defined service functions. The following scenario that includes a user's intention to use a service is typical of the cases that we used.

"I have come with my family by bus to a department store for shopping. Before shopping, we would like to get some lunch. After checking the timetable of the return bus, I would like to search for a restaurant near by the department store. I'll check today's information on sales at the department store on the way".

Using cases such as this one, we can extract user problem-solving requests that arise at various real-world locations as well as task knowledge associated with information services. Moreover, individual

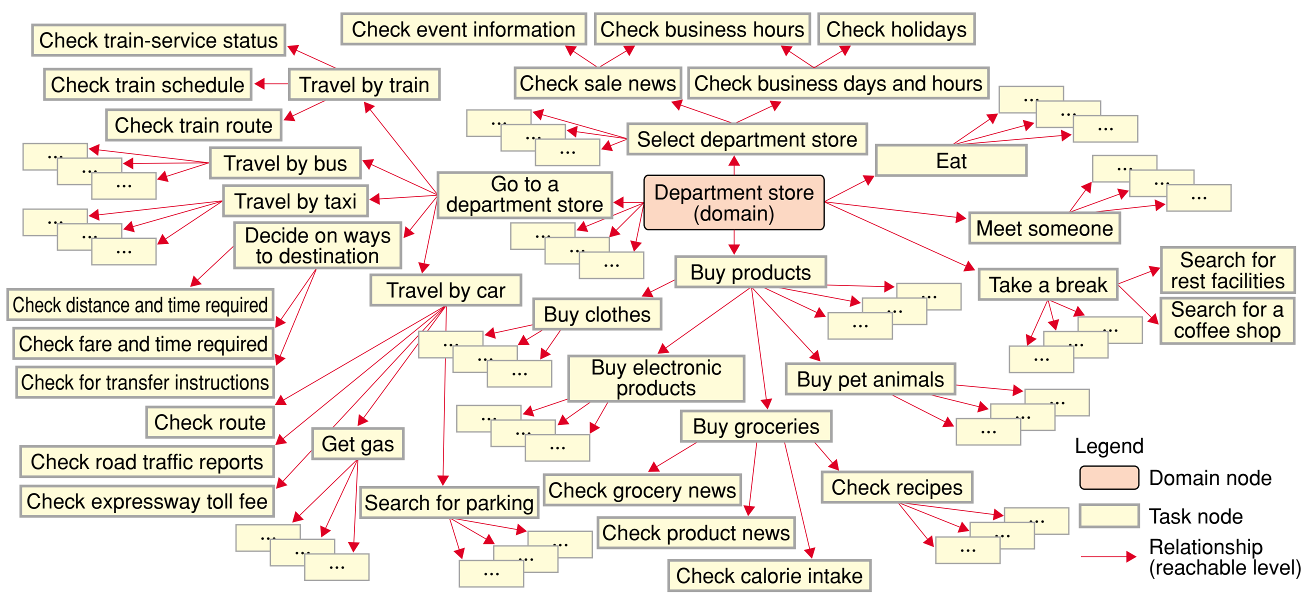

Fig. 3. Example of constructed task knowledge.

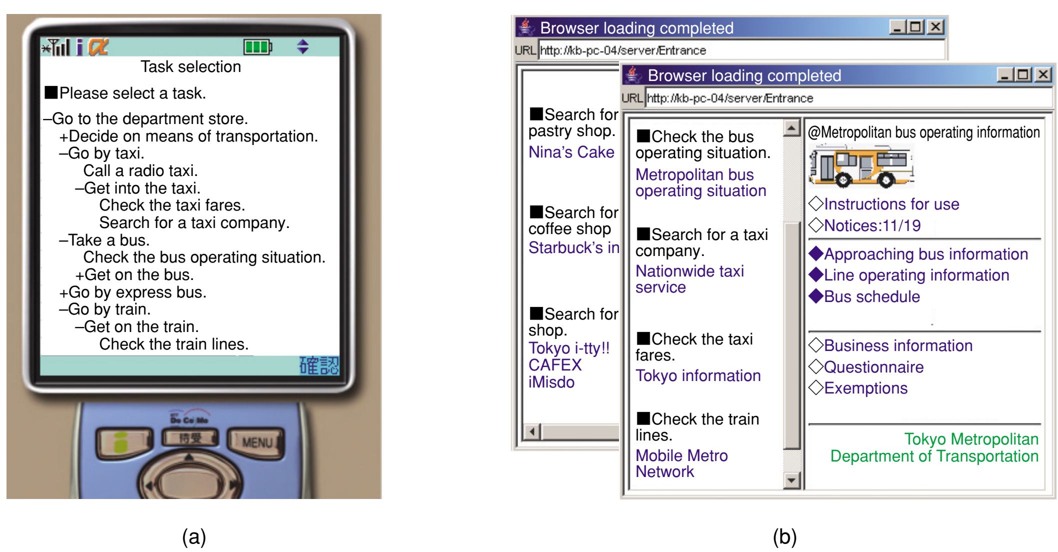

Fig. 4. Prototype screen shots.

cases can be easily collected using no more than the service-usage log, which prevents bottlenecks from occurring when this knowledge is acquired. About 500 cases have so far been collected targeting nine domains such as amusement parks and department stores. These cases are being applied to the construction of a task knowledge base. An example of a task knowledge base is shown in **Fig. 3**.

3. Implementation

Some typical screen shots (user interface) of a prototype version of the proposed service navigation system are shown in **Fig. 4**. The client application on the mobile terminal is implemented using a Java*1 emulator for mobile phones, and the knowledge server is implemented as a servlet*2 using the Tomcat application server running on Linux OS.

In this implementation, the user enters a problem-

*1 Java: an object-oriented development environment for networks promoted by Sun Microsystems, USA.

*2 Servlet: a module of Java code that runs in a server application to answer client requests.

solving request by means of text input. The system assumes that this input consists of words like “train” and “department store” and that the problem can be expressed in a natural sentence such as: “I would like to go to a department store” and “I would like to buy a DVD player”. When it receives the input of such a natural-language sentence, the system divides it into separate words by morphological analysis and creates multiple word sets for processing.

The knowledge server selects from the task knowledge base a task unit conforming to the problem-solving request and passes it to the client application. At this time, the knowledge server obtains from the service knowledge base the URIs of information services associated with the selected task unit and passes them as well to the client application. The client application then expands the received task unit and displays it in tree form (Fig. 4(a)). The user inspects the displayed task unit and selects one or several tasks that suit the current objective. Then, he or she selects a specific information service from those associated with the selected task(s). Now that an information service has been decided upon, the client application displays that network information service in an interactive viewing format. The current prototype system displays this information service in frame-type HTML (hypertext markup language) format using the emulator’s browser function (Fig. 4(b)).

4. Evaluation

We performed a subjective evaluation using ten subjects (four men and six women aged between 20 and 39) to examine the effectiveness of the proposed system. This evaluation was limited to the scope of the current knowledge base, i.e., to the currently targeted domains. Subjects were asked to compare the proposed system with keyword-based full-text searching and directory-based searching. The idea here was to evaluate the process up to the stage of finding a service for problem-solving purposes in terms of process functionality and comprehension (ease of understanding). Subjects were presented with a definition of a problem and procedures for solving that problem using each of the three systems. (The defined problem was solvable using any of them.) Specifically, the procedure for entering keywords and searching through a list of search results was presented with respect to a keyword-based full-text search engine and that for searching through a directory was presented with respect to a directory-based search menu. Problem-solving procedures were presented beforehand to minimize the effects that user experience with any of these systems might have on evaluation results. After the subject had solved the defined problem according to the problem-solving procedures, he or she was asked to score each system in the range from 1 (easy) to 5 (difficult). Three problems were prepared for each of two domains (amusement park and department store) for a total of six problems. Evaluation results are shown in **Table 1**.

Examining these results, we see that the proposed system had the best scores indicating that it is indeed effective. Furthermore, on comparing its results for the two domains targeted by this evaluation, we see that the scores for the department store domain were somewhat better than those for the amusement park domain. This suggests that the proposed system can function effectively for domains that have few sites (portals) that handle related information in a comprehensive manner.

Table 1. Possibility of reaching an appropriate service.

	Task	Average score for keyword-based search	Average score for directory-based search	Average score for proposed system
Amusement park domain	Decide on destination, check attractions, confirm airplane reservation, reserve hotel room	2.1	2.8	2.5
	Check traffic reports, check expressway route	1.5	2.3	2.6
	Check postal code	1.0	4.6	1.2
Department store domain	Decide on destination, check for sales, check train route, check weather	3.0	4.6	2.2
	Check route by automobile, check toll fees	1.9	3.0	1.6
	Search for nearby shops	4.3	2.1	1.5
Average		2.3	3.2	1.9

5. Conclusion

In this article, we proposed a service navigation system based on task knowledge and described a method of constructing a task knowledge base. We also described the implementation and evaluation of a prototype system and confirmed the effectiveness of the proposed system for a limited number of domains.

The proposed system organizes information services from the viewpoint of users and aims to enable even users who are unfamiliar with the use of information services to access and utilize required services. This system features a scheme for dividing an upper-level task into subtasks and displaying that task and its subtasks. The user is therefore presented with means of solving problems that he or she was not initially aware of, resulting in a "discovery" effect.

In the future, we aim to establish a system for efficiently expanding the knowledge base. We also plan to study a system for optimizing task knowledge by linking with various types of context information such as location information and sensor information from peripheral devices. We also plan to conduct a large scale user test.

References

[1] M. B. Eisenberg and R. E. Berkowitz, "Information Problem Solving: The Big Six Skills Approach to Library and Information Skills Instruction," Ablex Publishing Corp, Norwood, NJ, USA, 1990.

[2] T. Naganuma and S. Kurakake, "A task oriented approach to service retrieval in mobile computing environment," AIA 2005, 2005.

[3] A. Bandula, "Self-regulation of motivation and action through internal standards and goal systems," Goal Concepts in Personality and Social Psychology, A. P. Lawrence, pp. 19-85, Lawrence Erlbaum Assoc, 1989.

Takefumi Naganuma
Research Engineer, NTT DoCoMo Network Laboratories

He received the B.E. in industrial and systems engineering from Tokyo Institute of Technology, Tokyo in 1996. He joined NTT DoCoMo in 2002. He is currently engaged in research of mobile computing and knowledge processing. He is a member of the Information Processing Society of Japan and the Japanese Society for Artificial Intelligence.

Shoji Kurakake
Executive Research Engineer, NTT DoCoMo Network Laboratories

He received the B.E. and M.E. degrees in mathematical engineering and information physics from the University of Tokyo, Tokyo in 1983 and 1985, respectively. He joined NTT DoCoMo in 1985. He is currently engaged in research of text recognition, image analysis, video media handling, seamless applications, and ubiquitous services. He is a member of the Institute of Electronics, Information and Communication Engineers, IEEE, and the Association of Computing Machinery.