BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **13**, Special Issue

Sofia • 2013 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2013-0045

A Case Study of Question Answering in Automatic Tourism Service Packaging

Liu Wang, Lejian Liao*, Kai Yang**, Hai Tan****

** School of Computer Science, Beijing Institute of Technology, Beijing, PRC, 100081*
** Beijing Key Lab of Intelligent Information, Beijing, PRC, 100081*
*** Beijing Simulation Center, Beijing, PRC, 100081*
**** East China Institute of Technology, Nanchang, Jiangxi, PRC, 330013*
Emails: wangliu2000@163.com liaolj@bit.edu.cn yangkaigege@126.com htan@ecit.cn

Abstract: *While Web Question Answering System (WQAS) has made great progress in Internet currently, a major limitation of the information sources that current WQASs are using is limited to static page texts. Content information that many users-are interested in, is provided dynamically to Web through programs. Web services are becoming the dominant forms of such sorts of programs. In this paper, a novel model is proposed to address the problem of natural language answering based Web services. The expansions of OWL-S are used to describe the services, and predicate tuples retrieved from questions are matched to the services by PROLOG reasoning. To plan the services, SHOP2 planner is introduced in our architecture, which adopts a semantic Web for content modeling, logic grammar for question parsing, and Hierarchical Task Network (HTN) for user query problem solving. A case study in e-Tourism domain is investigated, which implements dynamic packaging for tourism products.*

Keywords: *WQA, Semantic web service, HTN, e-Tourism.*

1. Introduction

Current Web question answering systems mainly extract data from text information to answer the questions, limited to static text. As a kind of modular and self-expressive programs are described, published, located and revoked in Web, Web

Services can extract the underlying database information, provide a variety of methods of operation to obtain the real-time information. Increasingly rich Web service resources will gradually become the main way for obtaining dynamic information, such as flight services, weather services and so on. Their combination will be able to complete the more complex work of question answering.

However, user requirements are unlimited. When he asks "I want to travel to Sanya, please make a plan for me", he wants the system not only to provide flight and weather information, but also design tours for him. That is why AI planning is very important for the question answering system.

Tackling this issue, we proposed a formal model to express the question and the relevant workflow of services. In the application of realization, we introduce a Hierarchical Task Network (HTN) to do AI planning. While related Web services are mapped as parts of a planning domain, HTN planning tools SHOP2 [1] will make a task planning in the domain and composite related services to achieve the user's target. Therefore, the user can get some solutions to meet their dynamic needs.

E-Tourism industry is the best field where to apply this technology. In recent years, tourism industry is developing rapidly, and the survey shows that the tourism industry will grow by 200 % in 2002 [2]. More and more companies will release their travel information online, but users need to browse a variety of relevant Web pages to determine their travel plans. The more complicated the users' needs are, the more information must be searched for. Natural language is one of the easiest ways for a user to express what he wants regarding a service. The ideal system will arrange the travelling affairs automatically according to the user's requests, make a plan, or even book flights and hotels in advance. Service-based question answering we are proposing will be able to realize this idea, and it can get dynamic travel information, package all kinds of dynamic travel services according to the different needs of users, combining into a wealth of tourism products. The only thing users should do is to type in their thought (natural language), and press down the button to execute the program. Question answering provides the ability of automated process for tourism products. Web services sponsored by tourism companies or third parties provide the basis to access the dynamic information for question answering.

This paper puts forward a question answering layer architecture using Web services and the application of the e-Tourism. Through the semantic and natural language processing, the question and service will be described formally, mapped to the HTN domain, so that the system will transform question answering into a service-oriented task planning. At last, the sequence of services will be given; involving those you will perform the solutions to meet users' needs.

2. Related work

As the nature language processing is invited in question answering system, the user will get precise answers to a question instead of a related document. There are two ways to find the answers [3], one is for static data, such as documents, database and so on, the other is dynamic data, such as Internet resource. However, the magnitude of more Web Services are not considered, and some user questions should be carried out by a sequence control or functions, these goals will not be achieved by only finding the matching texts [4].

Service interface description languages, such as WSDL and related standards, are evolving rapidly to provide a foundation for interoperation between Web Services. At the same time, Semantic Web Service technologies, such as the Ontology Web Language for Services (OWL-S), are developing the means, by which the services can be given richer semantic specifications [5]. In [6] and [7], Semantic Web Services dynamic composition can be achieved by a HTN planner SHOP2 to compose Web Services which are described by semantic tag. This approach automatically composes Web services by the OWL-S and therefore lacks the ability to acquire the question and resolve the user's requirement. In [8] and [9], Templates are used to transform the user's question. The user requests are matched with the semantic service descriptions by the ontology and the lexical vocabulary, but it cannot be used in dynamical environments. The user's request is transformed into a semantic graph in [10] and [11], and semantic matching takes place to organize a workflow of the retrieved services. Their solution is not suited for a large scale application. In [12], the authors retrieve the relevant services by using concept trees created by WordNet lexicon. However, they cannot identify the parameters of actions.

3. Conceptual model architecture

In this section we propose a theoretic model of question answering based Web services, and realize it by the architecture design. Here the architecture will integrate NLP, semantic ontology, HTN, and change the answering into a planning problem based on services. The application will complete the dynamic packaging by using various tourism products exposed by the Web service, and provide users with automated travel planning services. Question and answering (Q & A) supports the user's natural language input, the answer to the user level will be the Web service execution sequence planned by HTN.

A common Web question answering system is based on information extraction or Web resources digging. As the natural language processing technology is integrated into the answering system, you can address the problem of natural language, and it can resolve the question and complete lexical, syntactic, and semantic mapping, which can give precise answers or some relative documents.

Taking into account, that the text is only a small part of Web resources, a large number of computing services are provided by Web services. While the user's questions are various and complex, a large number of questions need to refer to different sources, to be extracted from different information, and to be combined with other services. Through these Web services exposed, you can get a new way to solve the question problem and meet customers' various needs.

Taking tourism industry as an example, with traditional applications, people must visit a number of independent travel sites to plan their trips or vacations, register their personal information for multiple times, and spend hours or days waiting for a response or confirmation. Consumers are discouraged by the lack of functionality. The dynamic packaging can resolve this problem. It is defined as the combination of different travel components, bundled and priced in real time, in response to the request of a consumer. But this approach does not comply with the highly dynamic and centralized nature of tourism industry [13].

Introducing question answering using Web services to the field of tourism can achieve a more flexible feature, and customers can enter a natural language description of the idea to explain their request. According to the input requirements, the system can search and match the relative services in a number of Web services provided by the travel companies, and make a real-time planning to generate a sequence of Web services, namely, tourism solutions. Clicking OK to implement the program, users can complete a series of processes, such as book flight tickets, reserve a hotel, and order a car, and so on to achieve their own travel goals.

For this purpose, this paper proposes a concept model of question answering based Web services. We use two-tuples to describe the users' question, $Q(\Phi, \Psi)$ [14], and the answer will be described as $A(S, \Psi)$, which is a services sequence S satisfied with Ψ.

Φ denotes restrictive sets:

$\Phi=\{\varphi_1, \varphi_2, \varphi_3, \ldots, \varphi_n\} \mid n=1, 2, 3, \ldots;$

Ψ denotes target sets:

$\Psi=\{\psi_1, \psi_2, \psi_3, \ldots, \psi_n\} \mid n=1, 2, 3, \ldots;$

S denotes services sequence:

$S=\{s_1, s_2, s_3, \ldots, s_n\} \mid n \le m.$

It is supposed that there are n services from m available, $n \le m$, and the system will make planning in m services, then we get a services sequence consisting of n services. Generally, we may use the triple $P=(\sum, s_0, s_g)$ to describe a planning question, among which $\sum=(s, a, \gamma)$ is a planning domain, s is a states set, a is an actions set, γ is a certain type state transition function, s_0 is the initial state of the system, and s_g is the target state of planning.

The following Fig. 1 shows the question answering layer architecture using Web services. The **System layer** provides the system support, such as an operating system, database, and communication protocol. The **Services layer** provides various Web services for all kinds of functions. The **Presentation layer** includes services description, management and discovery. The **Interpretation layer** supplies interfaces to complete language parsing and semantic parsing, and subsequently generate the initial state description s_0, the target description s_g and the planning domain description $\sum$ (functions, operations and methods, etc.). These descriptions will be mapped into related domains in the planning layer. The **Planning layer** is responsible for planning and dynamic packaging, and the HTN planner SHOP2 will be used to achieve that goal. The **Question answering layer** provides the user interface with domain applications. Between the question answering layer and services layer, the **semantic layer** and **rules layer** give the supports of semantic and reasoning.

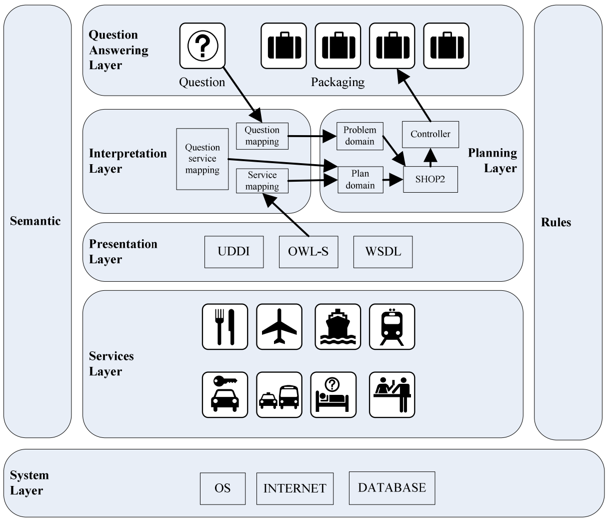

Fig. 1. Layer architecture of question answering

4. Implementation details

4.1. Question interpretation

Natural Language Processing (NLP) will be used in our model for question understanding of Web Question Answering (WQA). By using PROLOG, the natural language grammars can be written, and semantic descriptions that use logic formalisms are produced. After that, we can express them with the First Order Logic (FOL), and make some reasoning based on ontology [15].

We used DCG to parse the character list into a word list. And then choose verb-noun couples as target sets (Ψ), choose the numbers and dates as restrictive sets (Φ). This word list will be transformed into a predicate. For example, we can transform "I want to rent a car this Friday" into *rent*(*car*) and *date*(*Friday*). Therefore, the user's sentences can be represented as predicates [14], the natural language descriptions will be translated into $Q(\Phi, \Psi)$.

For example, "I want to buy a plane ticket to Shanghai from Beijing at 2013-2-10" will be put into a word list, and changed into a command couple, such as [buy, plane ticket]. After that, the couple can be represented as a target set *buy*(*plane ticket*) for problem mapping, and exported to the planning domain, described in Section 4.4.

Some restrictive sets will be used to property mapping or services mapping, such as *to*(*Shanghai*), *from*(*Beijing*) *and Date*(*2013-2-10*). The mapping method will be discussed in Section 4.3.

After processing the user's question with NLP, the Semantic Web matching module will search and discover related Semantic Web Services from UDDI, such as *searchCarService*, *bookCarService*, *searchHotelService* and *bookRoomService*, etc. These services will be transformed into HTN planning domain subsequently [6].

4.2. Semantic resolving

OWL-S is used to describe semantic information of Web Services. The OWL-S ontology has three main parts: the service profile, the process model and the grounding. The service profile is used to describe what the service does. This information includes the service name and description, limitations on applicability and quality of service, publisher and contact information. The process model describes "how to do". This description includes the sets of Inputs, Outputs, Pre-Conditions and results of the service Execution (IOPEs). The service grounding specifies how to access services. Extending the Effects (results of the service execution) of the process model will help us to select service operations that satisfy the users' requests and needs [8].

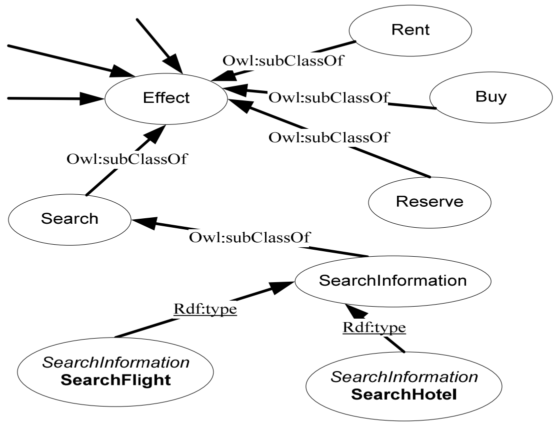

Fig. 2. Effects extension

Fig. 2 extends the Effects ontology, which describes Web Service's function and efficiency. It is used in Web Service searching to discover the Web Service that would fulfill a specific need within some quality constraints. It plays a key role in reasoning as described in Section 4.3.

Three components of OWL-S build a complete description of semantic Web Services. The relative OWL-S description will be resolved for formal description as follows:

```
(:operator
(!bookFlight ?flight number ?date ?type)
((have ? flight number)  (have ?date) (have ?type) )
()
(have ? plane-ticket)
)
```

These kinds of formal descriptions of Web Service formation will be exported to a part of the planning domain introduced in Section 4.4.

4.3. Question service mapping

Semantic ontology plays a key role in question services mapping. Travel domain ontology will be presented in Protégé. The following text shows Accommodations, and Guestroom, etc., which are quite normal for a travel domain [16].

```
<owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Accomodation">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:ObjectProperty rdf:ID="hasRoom"/>
        </owl:onProperty>
        <owl:someValuesFrom>
          <owl:Class rdf:ID="Guestroom"/>
        </owl:someValuesFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf...
```

With the help of travel domain ontology and semantics in the Effects Marks of OWL-S mentioned in Section 4.2, the reasoning will be handled by PROLOG. Through reasoning, the formal description of the question will be mapped to the Web Services described with OWL-S, given in Fig. 3 [15].

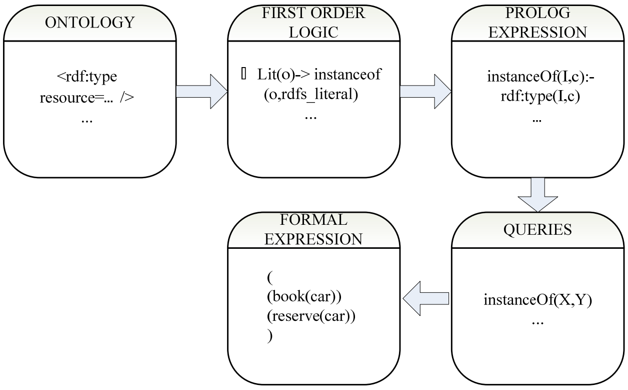

Fig. 3. Question service mapping flow chart

In order to get a formal representation of mapping, ontology is firstly represented by First Order Logic, next represented as a PROLOG computing formula. Here SWI-PROLOG [17] is used as a compiler. The mapping status is tested through various queries. Finally, the successful mapping rules as given below will be expressed formally to question service mapping, and exported to the planning domain in Section 4.4:

```
(
(book(car))
(!reserveCar ?cartype ?date)
)
```

4.4. HTN planning

In HTN planner SHOP2, the Planning problem is represented by a triple *P*(*S*, *T*, *D*), where *S* is the initial state, *T* is a task list and *D* is a domain description [7]. With these inputs, SHOP2 will return a plan *P* which is a sequence of operators that will be executed to make the state transit from *S* to *D* and achieve *T*.

We will use a HTN planning system SHOP2 to solve our problem. The objective of HTN is to produce a serial of actions which perform a specific function or task. For this sequence of actions, we will get a HTN problem domain and a planning domain for Web service dynamic composition. The problem domain includes task targets and rules, and the planning domain includes operators (like atomic functions) and a set of methods which denotes how to decompose a task into subtasks.

In Section 4.1 the description of the problem is generated to accomplish problem mapping, and the predicate commands of natural language will be mapped to the problem domain. In Section 4.2, semantic Web services are described as service mapping transformed into HTN planning domain. In Section 4.3, the mapping rules of question and service are generated by question service mapping, establishing the link between the problem domain and the planning domain. Due to the limited space, we do not discuss the many mapping problems that may occur in the problem service mapping part.

Thus, the question and services are mapped into the HTN problem domain and planning domain. By using the SHOP2 planner, the dynamic packaging products based on semantic Web services can be fulfilled, and the solution to the user's question can be obtained in Web question answering system.

5. Application and results

Here we describe the whole process of the system to explain how to satisfy the customers' demands by using question answering based on Web services. The process starts from the input of the user's question and does not end until the user gets the solution composed by a sequence of Web services.

Scenario. "I want to fly to Sanya from Beijing at 11/1 and fly back at 11/4, please give me a plan, including flight, car renting, hotel and view spot."

Users enter the above requirements through the Web client. These words contain many irrelevant words to the services discovery and imply that more than one service must be composed. These phrases will be handled by PROLOG for a natural language processing. According to the rules and grammar analysis, the logical description should be transformed to be suitable for SHOP2 domain (problem domain). Then the related Semantics Web Services will be discovered in UDDI, and OWL-S descriptions of Semantic Web Services will be translated into SHOP2 operators and methods. Reasoning based on domain ontology will be handled to complete the mapping between the problem and services. After that, a full SHOP2 planning domain will be built. SHOP2 planner starts to compute the planning problems, generate valid packages from the products offered by airlines, car rental, hotels, tourism companies, and organize a travelling plan. At last, the answer is sent back to the client as shown in Fig. 4. Each product corresponding to an appropriate Web Service will be packaged into a tourist composition service to provide users with a single quotation and discount. If the user executes the order, the controller will involve the Web Service sequence and automatically complete a series of operations, such as booking and paying bills. The tourism companies could use these packaged travelling products to broaden markets and attract consumers. Preliminary experiments have shown that the system efficiency of reasoning and planning is fair. Detailed statistics of subsequent experiments will be given in future work.

6. Conclusion and future work

We have described the architecture of question answering system using Web services, which is combined by the WQAS and the Semantic Web Service composition supported by HTN and NLP. PROLOG is used for NLP in question mapping, and by domain ontology and extended effects ontology in OWL-S, PROLOG does some reasoning to question service mapping. Furthermore, this system is introduced to E-Tourism. This gives the customers more flexible travel solutions through the question answering portal. In summary, using Web services provides us with a new way to Q&A to meet more changeable customers' requirements. In E-Tourism, based on the architecture, travel companies can extend their individual Web services, and provide more dynamic packaging products to satisfy client's requirements.

For future researches, interactive question interface will be studied to improve planning answering. Web service involving mechanism should be optimized for dynamic composition.

Fig. 4. Web question answering portal for tourism packaging

Acknowledgments: This work was supported by the Beijing Natural Science Foundation (Grant No 4092037).

References

1. Wu, D., E. Sirin, J. Hendler, D. Nau, B. Parsia. Automatic Web Services Composition Using SHOP2. – In: ISWC, LNCS, Vol. **2870**, 2003, 195-210.
2. Tariq, M., V. Kumar, S. Khoja, B. Chowdhry, M. K. Khan. Smart Travel Service Advisor Using Semantic Web and Agent Technology. – Computers and Simulation in Modern Science, Vol. **1**, 2008, 126-131.
3. Kaisser, M. Web Question Answering by Exploitingwide-Coverage Lexical Resources. – In: Proceedings of the11th ESSLLI Student Session, 2006, 203-213.
4. Dumais, S., M. Banko, E. Brill, J. Lin, A. Ng. Web Question Answering: Is More Always Better? – In: 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'2002), 2002.
5. Martin, D., M. Paolucci, S. McIlraith, M. Burstein, D. McDermott, D. McGuinness, B. Parsia, T. Payne, M. Sabou, M. Solanki, N. Srini-vasan, K. Sycara. Bringing Semantics to Web Services: The OWL-S approach. – In: 1st Int. Workshop on Semantic Web Services and Web Process Composition (SWSWPC'2004), San Diego, CA, 2004.
6. Sirin, E. Combining Description Logic Reasoning With AI Planning for Composition of Web Services. Ph. D. Dissertation, University of Maryland, College Park, 2006, 1-239.
7. Sirin, E., B. Parsia, D. Wu, J. Hendler, D. Nau. HTN Planning for Web Service Composition Using SHOP2. – Journal of Web Semantics, Vol. **1**, 2004, No 4, 377-396.
8. Bosca, A., F. Corno, G. Valetto, R. Maglione. On-the-Fly Construction of Web Services Compositions From Natural Language Requests. – JSW, Vol. **1**, 2006, No 1, 40-50.
9. Bosca, A., A. Ferrato, F. Corno, I. Congiu, G. Valetto. Composing Web Services on the Basis of Natural Language Requests. – In: IEEE International Conference on Web Services (ICWS'05), 2005, 817-818.