A Web services-based and Ontology-supported Case-Based Reasoning Information Agent

Sheng-Yuan Yang
Department of Computer and Communication Engineering, St. John’s University, Taiwan
499, Sec. 4, Tam-King Rd., Tam-Shuei, New Taipei City 25135, Taiwan, ROC
yay@mail.sju.edu.tw

Abstract—In this paper, a case-based reasoning information agent with Web service and ontology techniques was presented. The preliminary system developments and tableaux show that the research results not only attest to the feasibility of the proposed architecture, but are also highly successful.

Keywords—Web Services, Ontology, Case-Based Reasoning Agents, Cloud Information Systems

I. INTRODUCTION

Using cases is an important method for acquiring knowledge. Case-based reasoning, CBR, is a problem-solving technique employing previous experiences and past successful cases for solving current problems. In the literatures, there are two examples of the development of information-related systems with case-based reasoning and learning techniques, including, Krause [2] and Gu et al. [1]. In short, the abovementioned researches employed traditional case-based reasoning techniques and related similarity measured mechanisms, but added improved and developed novel concepts by themselves, thereby stressing the importance of developing mechanisms by themselves. The proposed system also developed related similarity measure mechanisms. That is to say, going through the Web service CBR_InsTmpCaseData provided by WIAS (Web-service-based Information Agent System) [7] records, the sequence of monitor data is put into the “TmpTable”; the agent then calculates the case similarity compared with the field “Sensor Data” of cases in the system Case Base, finally going through the WIAS’ Web service CBR_Solutions provides the web services case retrieval, case reuse, case adaptation and case retaining to easily complete a case-based reasoning and learning mechanism. By way of the related system tableaux (described later), the testing outcomes can still put greater stress on the specific similarity measure mechanism to be more suitable for application domains.

Professor Jean-Lien Chen, President of St. John’s University in Taiwan, leads a research team focused on an enhanced energy project for a private school: Building an Intelligent YOHO Environment Based on Cloud Computing Structures. The related sub-projects include: 1) Constructing an Information Platform with U-live Intelligent Environment with a Hybrid Network; 2) The Study of an Image Recognition and Analysis System implemented in U-live Intelligence spaces; 3) Developing an Intelligent Green Energy Management System; 4) Design of an Intelligent Space Power and Luminance Management System; 5) The Study of an Intelligent Energy Management and Security & Anti-disaster System; and 6) Developing a Cloud Multi-agent system with Web Service and Ontology techniques for Intelligent YOHO Information Processing and Decision-making Support, six sub-projects in total. In regard to the whole project, this proposed system, i.e. the sixth sub-project, plays the role of a cloud intelligent multi-agent system with YOHO information processing, exchanging, communicating, operating, integrating, analyzing, and decision-making support in order to achieve an operation with precision, high speed, robustness, ubiquity and initiation. Fig. 1 illustrates the conceptual diagram of the proposed system.

Fig. 1. Role structure of the proposed system in the whole project.

Fig. 2 shows the detailed system structure of the proposed system, in which users can transmit related information to the backend server information management system through the Ubi-IA (Ubiquitous Interface Agent). The Solution Finder provides YOHO solutions as the control center of the system. First, the system initiates the OntoDMA (Ontological Data Mining Agent) [6], which uses data mining and cluster classification techniques and provides support for the backend operation mode of the system beforehand, meaning that it actively provides profound, up-to-date and fast YOHO operational knowledge of the proposed system, or finds the predicted rules; furthermore, through the OntoCBRA (Ontological Case-Based Reasoning Agent) [8] service and case similarity computing for inducing the Case-Based Reasoning mechanism, the system actively acquires related intelligent YOHO knowledge to enhance management capability for providing user-oriented and domain-related information solutions; finally, if the two abovementioned

978-1-4799-5277-9/14 © 2014 IEEE
DOI 10.1109/IS3C.2014.14
Ontology is a method of conceptualization in a specific domain. It plays various roles in developing intelligent systems, including knowledge sharing, knowledge reuse, semantic analysis of languages, etc. Development of ontology for a specific domain is not yet an engineering process, but it is clear that domain ontology must include descriptions of explicit concepts and their relationships about a specific domain. We have outlined a principle construction procedure in [3]: following the procedure, we have developed ontology for the energy-saving domain. Fig. 3 shows part of the ontology taxonomy. The taxonomy represents relevant energy-saving concepts as classes and their relationships as a variety of links. The figure illustrates an isa link, which allows the inheritance of features from parent classes to child classes. The author has also used Protégé’s APIs (Application Program Interface) to develop a set of ontology services which work as the primitive functions to support information retrieval from the ontology, including transforming query terms into canonical ontology terms, finding definitions of specific terms in ontology, finding relationships among terms, finding compatible and/or conflicting terms against a specific term, etc. Especially, the proposed system checks for similar cases using the VRelationship in the ontology. TABLE I illustrates some examples of VRelationship for processing energy-saving related information.

### II. PROPOSED SYSTEM STRUCTURE

#### A. Domain Ontology as Fundamental Semantics

Ontology plays various roles in developing intelligent systems, including knowledge sharing, knowledge reuse, semantic analysis of languages, etc. Development of ontology for a specific domain is not yet an engineering process, but it is clear that domain ontology must include descriptions of explicit concepts and their relationships about a specific domain. We have outlined a principle construction procedure in [3]: following the procedure, we have developed ontology for the energy-saving domain. Fig. 3 shows part of the ontology taxonomy. The taxonomy represents relevant energy-saving concepts as classes and their relationships as a variety of links. The figure illustrates an isa link, which allows the inheritance of features from parent classes to child classes. The author has also used Protégé’s APIs (Application Program Interface) to develop a set of ontology services which work as the primitive functions to support information retrieval from the ontology, including transforming query terms into canonical ontology terms, finding definitions of specific terms in ontology, finding relationships among terms, finding compatible and/or conflicting terms against a specific term, etc. Especially, the proposed system checks for similar cases using the VRelationship in the ontology. TABLE I illustrates some examples of VRelationship for processing energy-saving related information.

#### B. System Architecture and Operations

![Fig. 2. The system structure of the proposed system](image)

Table I: Examples of VRelationships of Energy-Saving Related Information

<table>
<thead>
<tr>
<th>RELATIONSHIP</th>
<th>FEATURE</th>
<th>VALUE</th>
<th>EXPLANATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mutually exclusive</td>
<td>Season</td>
<td>Summer</td>
<td>One of the four divisions of the year cannot have a different value at the same time.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Winter</td>
<td></td>
</tr>
<tr>
<td>Downward-compatible</td>
<td>Temperature</td>
<td>26°C</td>
<td>A temperature value that contains the value of 26 - 28°C can be regarded as the one with 26°C.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>26-28°C</td>
<td></td>
</tr>
<tr>
<td>Conditionally</td>
<td>Cooler</td>
<td>Fan</td>
<td>A fan can be regarded as one type of cooler with lower energy consumption.</td>
</tr>
<tr>
<td>Downward-compatible</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 4 shows the structure of the OntoCBRA. The Case Generator is responsible for constructing case resources, and storage in the ODAC (Ontological Database Access Cases), which is based on Historical Information provided by the WIAS, and used as indices to structure those cases with the help of the system ontology. If the query information is whether CBR solutions exist, the Case Retriever gets the same or similar cases as candidate cases, and then calculates their degree of similarity. If the case is absolutely the same as the query, the Case Reuser directly returns its solution part to the Ubi-IA. If the case is not a completely identical situation, and reaches the pruning standard (the system defined threshold), the agent passes the case into the Case Reviser for further processing. The Case Reviser provides the most suitable pruning solution and returns it as a reference solution to the Ubi-IA. The system enters the step of Case Retain if the reference solution has a higher user satisfaction; otherwise the system directly discards the reference case. Finally, the Case Retainer decides whether to keep the revised case, based on its case similarity, and stores it in the ODAC, as detailed in [4]. The ODAC also provides Case Information to become information material for constructing Prediction Rules within
the OntoDMA. The system ontology is the domain fundamental semantics, while the Adaptation Rule Base provides the corresponding rules to the Case Reviser for producing adapted solutions.

In the three-stage intelligent decision processing strategy, the OntoDMA processes the monitor data one by one to find suitable prediction solutions; at the same time, the OntoCBRA only sequentially records related monitor data. When the OntoDMA cannot provide the corresponding feedback controlling response, the Solution Finder of the Ubi-IA triggers the OntoCBRA to provide profound and state-of-the-art feedback controlling responses. At this time, the OntoCBRA batch processes the sequence of monitor data to obtain CBR solutions. In other words, the case applying of the OntoCBRA is a processing operation of the corresponding feedback controlling responses, based on the on-line sequence of monitor data. This process mainly uses the help of corresponding Web services provided by the WIAS, including: 1) CBR_InsTmpCaseData: inserts temporal data to get back CBR solutions; 2) CBR_Solutions: finds CBR solutions by the case processing cycle, including: case retrieval, case reuse, case adaptation and case retaining (detailed descriptions in [8]).

### III. SYSTEM DISPLAY

As mentioned above, a case is the most usually occurring situation with its corresponding operational mode at a specific time slot. Given an example on the energy-saving information processing, firstly, the system transforms the fields of “MAC”, “MAC_Type_Name” and “Sensor_Data” of the original data into the case data of “Area_Name”, “Area_Type_Name” and “Area_Size_Name”, with space semantic based on the predefined rules of “MAC_Sensor” in the system database. In addition, the system extracts the frequent appearance records from the field of “Sensor_Data” (the most stable energy-saving operational plan in the specific monitor space) in accordance with the whole classification of the three items “Area_Name”, “Area_Type_Name” and “Area_Size_Name.” Finally, the system merges the fields of “Humidity”, “Temperature”, “Illumination” and “CO₂” into the two data columns “MAC_Type_Name” and “Sensor_Data” to complete the construction of the ODAC; at the same time, those cases are indexed by the system ontology, as shown in Fig. 5.

In the three-stage intelligent decision processing mentioned above, the OntoDMA can deal with abnormal monitor data; at this time, the OntoCBRA only starts the corresponding Web service CBR_InsTmpCaseData provided by the WIAS to record the sequence of monitor data into Case_TMP (which has already been transformed into semantic information). If the OntoDMA cannot process, the OntoCBRA can be triggered by the Solution Finder of the Ubi-IA and go through the Web service CBR_Solutions provided by the WIAS to start the on-line case application procedure. Given the same energy-saving example as mentioned above, the system can classify and sort all “MAC_Type_Name” in the specific space, and then calculate the corresponding “avgVALUE” and its “avgvRangeUp” and “avgvRangeDown” based on the part of “Area_Name”, “Area_Type_Name” and “Area_Size_Name.” For expanding the case extraction range, the value of “avgVALUE” is the system base, and when multiplied by 75% it is the value of “avgvRangeDown”, and by 125% it is the value of “avgvRangeUp”; it is used to avoid the awkward situation of retrieving nothing with the solo value of “avgVALUE” as the standard of case retrieval. In other words, the system employs the range from “avgvRangeDown” to “avgvRangeUp” to be the base of case retrieval, as shown in Fig. 6.

In case retrieval, the OntoCBRA proceeds with the sifting similarity threshold based on the range from “avgvRangeUp” to “avgvRangeDown,” with the help of the VRelationship in the system ontology. Fig. 7 illustrates all cases tallied with the retrieval threshold. If no case tallied with retrieval threshold exists, then the system does not have any experiences to deal with analogous situations, so the OntoCBRA returns the null value back to the Ubi-IA for future processing. The system then parses all those cases with the help of VRelationship in the system ontology, and calculates the similarities between the sequence of monitor data in “Case_TMP” and them, as shown in Fig. 8. Assume that “completely equivalent,” i.e., the column value of “Similarity”, is “1”, which means “100%” similarity; the system then directly outputs the case as the CBR solution, and sends it back to the Ubi-IA as the feedback controlling response to the sequence of monitor data in order to complete the step of case reuse. If “partly analogous but not completely equivalent,” the agent goes to the step of case adaptation.

![Fig. 5. Part of the content of semantic case base](image1)

![Fig. 6. Range of system case retrieval](image2)
When the OntoCBRA step mentioned above cannot successfully obtain completely equivalent cases, the agent will go to the step of case adaptation with the help of a combination of the VRelationship in the system ontology and corresponding adaptation rules, as shown in Fig. 9. The field “SimilarityCount” means the sum of the field “Similarity” with the same “Case_ID”; while the field “SimilarityAvg” equals their similarity averages; finally, the agent chooses the “Case_ID” with the highest ranking similarity degree to be the outcome of the case adaptation. The agent sends it back to the Ubi-IA as the feedback controlling response to the sequence of monitor data in order to complete the case adaptation step.

The OntoCBRA goes through the Web service CBR_Solutions to carry out the part of case adaptation; currently, the operation rule is based on the highest similarity ranking and its average. In the future, we will introduce the method ‘adapted by the system monitor staff’, i.e., the staff directly adjusts their Sensor_Data values of those cases with values lower than the threshold in the step of case reuse. The establishment of adjusting value can be set by domain experts externally. In the part of case retaining, the agent can directly calculate their corresponding similarities and survival values of those cases in the case adaptation step, according to the user satisfaction degree, and store them up to the system threshold in the ODAC, as shown in “Case 61” of Fig. 10. The establishment of system threshold can also be set by domain experts externally.

IV. CONCLUSIONS AND DISCUSSION

This paper proposed a cloud case-based reasoning information agent with Web service and ontology techniques. Not only can it explore related technologies in order to establish a Web service platform, but it can also study how to construct cloud interactive diagrams to employ Web service techniques for extensively and seamlessly integrating related information agents on the Internet. The preliminary system developments and tableaux show that the research results not only highlight the feasibility of the proposed architecture, but are also highly successful.

ACKNOWLEDGMENT

The author would like to thank Hung-Chun Chiang, Ming-Yu Tsai and Guo-Jui Wu for their assistance in system implementation and experiments. This partial work was supported by the National Science Council, ROC, under Grant NSC-102-2632-E-129-001.

REFERENCES


