Developing a Cloud Energy-saving and Case-Based Reasoning Information Agent with Web Service Techniques

Sheng-Yuan Yang
Department of Computer and Communication Engineering
St. John's University,
New Taipei City, TAIWAN
ysy@mail.stju.edu.tw

Abstract—Web service techniques are presented herein for supporting a cloud energy-saving and case-based reasoning information agent. Not only can it explore related technologies 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 energy-saving and a case-based reasoning information agent on the Internet. The preliminary system development, display and corresponding comparisons show that the research results can not only sketch the feasibility of the proposed architecture but also are highly successful. Furthermore, the system evaluation shows around 40% of the data queries can be answered by the proposed system, which can effectively alleviate the overloading problem usually associated with a backend server, and its correct rate of data solutions is around 85.1%, which is a very efficient improvement in data query response.

Index Terms—Web Services, Case-Based Reasoning, Cloud Information Systems, Energy-saving Information Systems.

I. INTRODUCTION

Using cases is an important method for representing knowledge. Case-based reasoning, CBR, is a problem-solving technique employing previous experiences and past successful cases for solving current problems. In other words, it employs reasoning rules based on the degree of case similarity to find a way to solve problems, and extends this problem solution to suitably solve new problems. In the view of cognitive science, those accumulated cases of the system are processing experiences which can be directed at the domain problem. When those experiences are abundant, they cover a bigger problem solving domain. In addition, all of the cases are indexed with real data by an ontology-like structure for fast and precise information accessing. Determining how to establish this case structure with CBR and then to support the whole system operation in order to robustly provide energy-saving information services are the main research and development points of the proposed system.

In the networking era, many Web services have resulted from developing the interface of traditional program providers with cloud computing techniques. Web Services principally provide services for application programs on the Web and enable the usage of programs in other machines which are provided with powerful inter-communication and extensibility functions. It can easily integrate application programs and related programs on the Web and achieve some complicated information service processes through interactive programs. Related standards contain XML, SOAP, WSDL and UDDI [5, 9]. It is ready to go through networks, transmit necessary service interfaces to needed programs, and even proposes formats of communication standards. When program scales need addition or modification, they can immediately be achieved through Web services. In regard to cloud computing environments, this paper developed a case-based reasoning information agent on the basis of Web service techniques, which can easily achieve the application goal of ubiquitously accessing energy-saving information.

Fig. 1. Conceptual architecture of an energy-saving information system

Due to the distribution of space and monitoring hosts, energy-saving information systems require a more flexible manner of program development. Replacing an existent application interface with functions of network transmission and Web services can not only easily achieve on-line and real-time addition/modification services, but also immediately extend its powerful functions. Fig. 1 illustrates the conceptual architecture of an energy-saving information system. The left-hand side was constructed with a wireless sensor network to detect and collect the running parameters of all of the electrical devices; then the related data would be sent to a cloud server for related energy-saving information processing and corresponding decision support [8]. The cloud server is a multi-agent system [9], including: Interface Agent [11], Data Mining Agent [12], Case-Based Reasoning Agent (CBR Agent)
[10], and Web-Service-Based Information Agent System (WIAS) [13], as shown in Fig. 2. The Interface Agent is responsible for providing energy-saving monitoring of information access and intelligent decision making. The latter is aimed at providing corresponding control decisions to monitor information, including whether prediction solutions exist, as judged one by one by the Data Mining Agent; whether CBR solutions exist, as a series of judgment by the CBR Agent; and whether predefined solutions exist, as judged one by one by the Interface Agent in accordance with predefined rules within WIAS; this is called three-stage intelligent decision processing. WIAS exploits the concept of SQL IC to provide various Web services with energy-saving information from the abovementioned agent systems, which can achieve the investigations regarding fast accessing system information in clouds via the Internet.

![Fig. 2. System structure of backend multi-agent system](image)

In summary, the paper focuses on developing a cloud energy-saving and case-based reasoning information agent with Web service technique. Not only can it explore related technologies 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 energy-saving and a case-based reasoning information agent on the Internet. The preliminary system development, display and corresponding comparisons show that the research results can not only sketch the feasibility of the proposed architecture but are also highly successful.

II. BACKGROUND AND TECHNIQUE

A. Cloud Computing

Cloud computing is a technique of Internet-based development and use of computer technology. In other words, it will set up the necessary operating resources and related data on the Internet so that users can directly use them when they access the Internet. Furthermore, determining how to construct cloud interactive diagrams for extensively and seamlessly integrating energy-saving and case-based reasoning information agents through Web service technique is also the investigation focus and development point of this study.

B. Structure of WIAS

Fig. 3 is the structure of the Web-service-based Information Agent System, WIAS [13]. If the query information is an access command, WIAS directly goes through the Web-Service-Based Interface to employ the SQL IC Constructor to trigger the corresponding SQL access templates. After binding related access parameters, WIAS retrieves the corresponding access results from the Raw Data Base. Finally, it goes through the Web-Service-Based Interface to return those results to the Interface Agent. If the query information is whether predefined solutions exist, WIAS also goes through the Web-Service-Based Interface to ask the Predefined Rule Base to return the corresponding predefined solutions to the Interface Agent. Furthermore, the Raw Data Base also provides all of the frequent historical information to the CBR Agent as information material for the production of cases, and supplies all infrequent historical information into the Data Mining Agent as information material to trigger the production of prediction solutions. As for the construction of Predefined Rules, it is determined by domain experts through the Rule Maker.

![Fig. 3. Structure of WIAS](image)

(b) SQL IC

Fig. 4. Conceptual architecture of Web services in WIAS
The WIAS architecture realizes on-line interfaces of Web services with cloud computing and information transmission techniques on the Internet. It enables individual agents to access the common function library, as shown in Fig. 4 and part of function descriptions in TABLE 1, in order to immediately respond to the corresponding agent that uses this Web service. Not only can it easily achieve communication with the backend database, but it can also easily add and renew related functions for the research purpose of designing a Web-service-based information agent with clouding techniques.

### TABLE I. PART OF WEB SERVICE DESCRIPTIONS

<table>
<thead>
<tr>
<th>SERVICE NAME</th>
<th>SERVICE DESCRIPTION</th>
<th>TO WHOM</th>
</tr>
</thead>
<tbody>
<tr>
<td>CBR_InitTempData</td>
<td>Inserting temporal data for CBR solution</td>
<td>CBR Agent</td>
</tr>
<tr>
<td>CBR_Solution</td>
<td>Finding CBR solutions</td>
<td>CBR Agent</td>
</tr>
<tr>
<td>CBR_TransToCone</td>
<td>Transforming raw data into geometric case</td>
<td>CBR Agent</td>
</tr>
<tr>
<td>CBR_VarySensorInfo</td>
<td>Viewing sensor information</td>
<td>CBR Agent</td>
</tr>
<tr>
<td>CBR_VarySensorValue</td>
<td>Viewing modified rules</td>
<td>CBR Agent</td>
</tr>
<tr>
<td>DB_InCaseBase</td>
<td>Inserting case-base rules</td>
<td>Sharing</td>
</tr>
<tr>
<td>DB_InPredication</td>
<td>Inserting prediction rules</td>
<td>Sharing</td>
</tr>
<tr>
<td>DB_InSyncBase</td>
<td>Recording if the case has been retrieve</td>
<td>Sharing</td>
</tr>
<tr>
<td>DB_LoadData</td>
<td>Recording if the raw data has been transformed</td>
<td>Data Mining Agent</td>
</tr>
<tr>
<td>DM_Solutions</td>
<td>Finding predefined solutions</td>
<td>Data Mining Agent</td>
</tr>
<tr>
<td>DM_TransCaseToPred</td>
<td>Transforming semantic cases into prediction rules</td>
<td>Data Mining Agent</td>
</tr>
<tr>
<td>DM_VeriBase</td>
<td>Verifying case-base</td>
<td>Data Mining Agent</td>
</tr>
<tr>
<td>DM_VeriPrediction</td>
<td>Verifying prediction rules</td>
<td>Data Mining Agent</td>
</tr>
<tr>
<td>DM_VeriRuleBase</td>
<td>Verifying predefined rules</td>
<td>Data Mining Agent</td>
</tr>
<tr>
<td>DM_VeriRuleData</td>
<td>Verifying raw data</td>
<td>Data Mining Agent</td>
</tr>
<tr>
<td>DM_Solutions</td>
<td>Finding predefined solutions</td>
<td>Interface Agent</td>
</tr>
<tr>
<td>Data_BdUniqueTables</td>
<td>Deduplication in each individual rules</td>
<td>Sharing</td>
</tr>
<tr>
<td>Data_D_Numeric</td>
<td>Deduplication of variable’s not an integer</td>
<td>Sharing</td>
</tr>
<tr>
<td>Data_VeriCDB</td>
<td>Verifying CDB</td>
<td>Sharing</td>
</tr>
<tr>
<td>Data_VeriPred</td>
<td>Verifying predefined rules</td>
<td>Sharing</td>
</tr>
<tr>
<td>Data_VeriRules</td>
<td>Verifying rules</td>
<td>Sharing</td>
</tr>
<tr>
<td>Data_VeriData</td>
<td>Verifying predefined rules</td>
<td>Sharing</td>
</tr>
<tr>
<td>Data_VeriSensorInfo</td>
<td>Verifying basic information of the specific sensor</td>
<td>Sharing</td>
</tr>
<tr>
<td>Data_VeriUniqueTables</td>
<td>Verifying duplicate data in each individual rules</td>
<td>Sharing</td>
</tr>
<tr>
<td>Data_VeriValueTables</td>
<td>Verifying duplicate data in each individual rules</td>
<td>Sharing</td>
</tr>
</tbody>
</table>

### C. Developing Techniques

The system prototype adopted MS SQL Server and My-SQL Server as the sharing platform of the backend databases; the information safety can be guaranteed by different servers with mutual backup information. SQL is a query language for obtaining data from the relational database. This paper employed the concept of SQL access templates to construct the usual SQL IC with C#. Their functions are like those of the IC’s, which can bind different parameters to easily access corresponding query results. Not only can this approach expand the Web service functions of WIAS, but it can also provide various query services of energy-saving information to other agents within the system. The rest of the agent was developed by Java and colocated in the multi-threading building manner, easily leading to a double-win strategy for multi-tasking processes and enhancing system performance.

### III. PROPOSED SYSTEM STRUCTURE

Fig. 5 is the structure of the CBR Agent [10]. The Case Generator is responsible for constructing case resources and storage in the Case Base, which is based on Historical Information provided by WIAS. If the query information is whether CBR solutions exist, the Case Retrieve 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 Reuse directly returns its solution part to the Interface Agent. If the case is not a completely identical situation and reaching the pruning standard (the system defined threshold), the agent passes the case into Case Revise for further processing. Case Revise provides the most suitable pruning solution and returns it as a reference solution to the Interface Agent. 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 Retain decides whether to keep the revised case based on its case similarity and stores it into Case Base, as detailed in [7]. The Case Base also provides Case Information to become information material for constructing Prediction Rules within the Data Mining Agent. Detailed operations and adopted techniques are described below.

### A. Cases Production

![CBR Agent Diagram](image)

Fig. 6. Production concept of Case (T: Temperature, C: CO2, and H: Humidity)

CBR is a problem-solving technique employing previous experiences and past success cases for solving current problems. For this reason, the proposed system gives a Case a definition: the most usual occurrence situation (that is most frequent happening) and its corresponding energy-saving operational mode at a specific time slot. Its ‘meaning’ is the most stable energy-saving operational plan in the specific monitor space. Fig. 6 is the production concept of cases. However, the original cases storage in Fig. 6 cannot be directly applied to various energy-saving operational modes. They have to be transformed into suitable semantic cases in accordance with the
corresponding MAC Tables; then they can be applied to the
mechanism for the reasoning of the majority of energy-saving
cases. Fig. 7 is the transformation concept of semantic cases
[10].

![Fig. 7. Transformation concept of semantic case](image)

**B. Cases Applying**

In the three-stage intelligent decision processing strategy,
Data Mining Agent [12] processes the monitor data one by one
to find suitable prediction solutions while the CBR Agent [10]
only sequentially records related monitor data. When the Data
Mining Agent cannot provide the corresponding feedback
controlling response, the Interface Agent [11] triggers the CBR
Agent to provide profound and state-of-the-art feedback
controlling responses. At this time, the CBR Agent batch
processes the sequence of monitor data to obtain CBR solutions.
In other words, the case applying of the CBR Agent 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 WIAS [13], including: 1) CBR_InsTempCaseData:
inserts temporal data to get back CBR solutions; 2) CBR_Solutions:
finds out CBR solutions by the case processing
cycle, including: case retrieval, case reuse, case adaptation and
case retaining. Related testing system tableaux and comparisons
are shown in Section IV.

**IV. SYSTEM DISPLAY, EVALUATIONS AND COMPARISONS**

The main system operations of CBR Agent comprise two
parts: periodical off-line cases production and immediate
on-line cases applying. As abovementioned, both can use the
help of Web services provided by WIAS for easily accessing
corresponding information services. Adapted techniques and
detailed operations are described below.

**A. System Tableaux**

As shown in Fig. 8, the materials of cases production
mainly come from the Raw Data Base of historical information
and the data quantum is pretty enormous. Therefore, if the
system directly employs one of MAC, MAC_Type and Data to
orderly carry out the matching process, this approach not only
costs the system process much time but also entails bigger
loading for the backend Data Base. For this reason, the system
introduces the concept of CBR and initiates the degree of case
similarity for directly applying solutions of existing cases. This
method can not only efficiently enhance the execution
performance of the whole system, but also easily solve the
system processing problems mentioned before. In other words,
the core task of the CBR Agent operation is to retrieve suitable
cases from the Case Base and efficiently employ those cases to
reduce the operation loading of the backend Data Base as well
as solve the analogy problems. Hence, cases production is an
important topic for discussion.

![Fig. 8. Content of part of the Raw Data Base](image)

**As abovementioned, a case is the most usual occurrence
situation with its corresponding energy-saving operational
mode at a specific time slot. 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, as shown in Fig. 9. In addition, the system
extracts the frequent appearance records in the field of
“Sensor_Data” (means 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 fields of “Humidity”, “Temperature”,
“Illumination”, and “CO2” into the two data columns:
“MAC_Type_Name” and “Sensor_Data” for finishing the
construction of the system case base, as shown in Fig. 10. This
approach presents the original and pure numeric cases in the
semantic manner based on the most usual occurrence situation
and its corresponding energy-saving operational mode at a
specific time slot. That is to say, it is the implementation of the
transformation concept of the semantic case in Fig. 7 mentioned.
above. Because the system is not meant to process single monitor datum but directly extracts a set of the best analogy cases based on cases similarities to be system solutions, this approach can easily solve the abovementioned problems.

situation on retrieving nothing with the solo value of “avgVALUE” as the standard of cases retrieval. In other words, the system employs the range from “avgvRangeDown” to “avgvRangeUp” to be the base of cases retrieval, as shown in Fig. 12.

To make use of on-line case applying, the CBR Agent can employ the standard of Predefined Rules in WIAS, which are constructed by Rule Maker through the help of domain experts, to compare with the monitor data. In the three-stage intelligent decision processing mentioned above, the Data Mining Agent can deal with the abnormal monitor data; at this time, the CBR Agent only starts the corresponding Web service: CBR_InstanceData provided by WIAS to record the sequence of monitor data into Case_TMP (which had already been transformed in semantic information), as shown Fig. 11. If the Data Mining Agent cannot process, the CBR Agent can be trigged by Interface Agent and go through the Web service: CBR_Solutions provided by WIAS to start the on-line cases applying procedure. Now, the system can classify and sort all “MAC_Type_Name” in the specific space and then calculate corresponding “avgVALUE” and its “avgvRangeUp” and “avgvRangeDown” based on the red part of three items: “Area_Name”, “Area_Type_Name” and “Area_Size_Name” shown in Fig. 11. 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” for avoiding the awkward

In cases retrieval, the CBR Agent proceeds with the sifting similarity threshold out based on the range from “avgvRangeUp” to “avgvRangeDown” (the red part of Fig. 12). Fig. 13 illustrates all cases tallied with retrieval threshold. If no case tallied with retrieval threshold exists, it means that the system does not have any experiences to deal with analogy
situations, so the CBR Agent returns the null value back to Interface Agent for future processing. The system then parses all those cases and calculates the similarities between the sequence of monitor data in “Case_TMP” and them, as shown in Fig. 14. Assume that “completely equivalent,” i.e., the column value of “Similarity” is “1” that means “100%” similarity, and then the system directly outputs the case as the system solution and sends it back to the Interface Agent as the feedback controlling response to the sequence of monitor data in order to complete the step of cases reuse. If “partly analogy but not completely equivalent,” the agent goes to the step of case adaptation.

When the CBR step mentioned above cannot successfully obtain completely equivalent cases, the agent will go to the step of case adaptation, as shown in Fig. 15. The field “SimilarityCount” means the sum of the field “Similarity” with the same “Case_ID” as shown in the red part of Fig. 14; while the field “SimilarityAvg” equals their Similarity averages; finally, the agent chooses the “Case_ID” with the highest ranking of similarity degree to be the outcome of the step of case adaptation, i.e., the “Case_ID: 267” as shown in the red part of Fig. 15. The agent sends it back to the Interface Agent as the feedback controlling response to the sequence of monitor data for complete the step of cases adaptation.

The CBR Agent goes through the Web service: CBR_Solutions to carry out the part of case adaptation; currently, the operation rule is based on the highest ranking of similarity 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 reverse the total degrees of similarity of those cases in the step of case adaptation, called case difference degree, and store them up to 75% into the system case base, as shown in “Case: 61” of Fig. 16. The establishment of case difference degree can be also set by domain experts from external.

### B. System Evaluations

![Fig. 17. 40 system cases produced by CBR_TransRawToCase](image)

We collected in total 348469 monitor data records from Mar. 1st, 2011 to May 31st, 2011 in the Electrical and Information Building, St. John’s University in Taiwan and then transformed those data records for storage in the system database. Our first experiment was to learn how well the CBR Agent works efficiently. We used those data records as the training data set. The Web service: CBR_TransRawToCase was used to construct in total 40 cases for storage in the system cases base, as shown in Fig. 17. We then randomly selected 5000 data query scenarios from the training data set as the testing data to test the performance of the CBR Agent. TABLE II illustrates the five-time experiment results that were achieved after comparing the returned solution one after another through domain experts. It shows, on average, out of 2018.4 queries, 805.6 queries can be taken care of by the CBR mechanism (around 39.9%); while leaving about 60.1% of the queries for the backend system to take care. In summary, around 40% of the data queries can be answered by the DMA, which can effectively alleviate the overloading problem usually associated with a backend server.

<table>
<thead>
<tr>
<th>Testing Order</th>
<th>Query</th>
<th>CBR Invocation</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2011</td>
<td>720</td>
<td>35.8%</td>
</tr>
<tr>
<td>2</td>
<td>1998</td>
<td>713</td>
<td>35.7%</td>
</tr>
<tr>
<td>3</td>
<td>1979</td>
<td>817</td>
<td>41.3%</td>
</tr>
<tr>
<td>4</td>
<td>2003</td>
<td>793</td>
<td>39.8%</td>
</tr>
<tr>
<td>5</td>
<td>2101</td>
<td>585</td>
<td>46.9%</td>
</tr>
<tr>
<td>Average</td>
<td>2018.4</td>
<td>805.6</td>
<td>39.9%</td>
</tr>
</tbody>
</table>

Our second experiment is to learn how well the CBR Agent works effectively. We used the same system cases base as the first experiment. We really collected in total 5000 monitor data records from Jun. 1st, 2011 to Aug. 31st, 2011 and then transformed those data records for storage in the system database. We conducted the experiment for five times. After each experiment, around 4–5 new cases are retained into the system cases base for the next experiment. TABLE III illustrates the five-time experiment results that were achieved.
after comparing the returned solution one after another through domain experts. It shows, on average, 40.1% of the data queries can be answered by the data-oriented CBR technique, while 85.1% of the correct rate of query solutions can be taken by the CBR Agent. In summary, the two experiments show, on average, 40% (\(39.9\% + 40.1\%\) / 2) of the data queries can be answered by the CBR Agent, leaving about 60% of the queries for the backend system to take care, which can effectively alleviate the overloading problem usually associated with a backend server; while its correct rate of data solutions is around 85.1%, which is a very efficient improvement in data query response to the brand new data.

<table>
<thead>
<tr>
<th>Testing</th>
<th>#Query</th>
<th>CBR Invocation</th>
<th>Correct Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order</td>
<td>%</td>
<td>%</td>
<td>%</td>
</tr>
<tr>
<td>1</td>
<td>1997</td>
<td>705</td>
<td>35.3%</td>
</tr>
<tr>
<td>2</td>
<td>2008</td>
<td>751</td>
<td>37.4%</td>
</tr>
<tr>
<td>3</td>
<td>2010</td>
<td>799</td>
<td>39.7%</td>
</tr>
<tr>
<td>4</td>
<td>1997</td>
<td>781</td>
<td>39.1%</td>
</tr>
<tr>
<td>5</td>
<td>2005</td>
<td>979</td>
<td>48.8%</td>
</tr>
<tr>
<td>Average</td>
<td>2003.4</td>
<td>803</td>
<td>40.1%</td>
</tr>
</tbody>
</table>

C. System Comparisons

In the literatures, there are a few examples on development of energy-saving-related information systems with case-based reasoning and learning techniques. For example, Chiang [1] proposed a maintenance management information system to provide an improving solution for the maintenance procedure and system structure of the Taichung Power Plant environment protection facilities control system, which is developed via the combination of electronic automation, protocol translation, case-based reasoning and relational database for reducing the control system trouble-shooting time to improve the reliability and performance of the equipments. Yang [6] selected the power industry as the research target and employed an Integrated Decision Tree and Bayesian Network method to integrate the opinions given in the expert meeting and build an analysis model for improving the prediction of electric power load and coping with the changing trend of the future. Wang [4] presented a case-based reasoning system to solve short term load forecasting problem with the aid of self-organizing maps and fuzzy-rough sets method, which used not only case-specific knowledge of past problems, but also used additional knowledge derived from the clusters of cases; it provided a new way for selecting proper feature subset and feature weights to improve the effectiveness of load forecasting. Niu et al. [3] proposed a novel load forecasting model that is a support vector model for load forecasting model based on case-based reasoning technology for solving the problems related to how to deal with the meteorological factor and the date class, improve forecasting accuracy and avoid artificial intervention. Yuan et al. [14] discussed what rough sets information entropy and principal component analysis are applied to the attributes reduction of load cases, such as decreased training time in the process of retrieval, as well as effective control implementation aimed at minor factors to essential ones for improving the effectiveness of short-term load forecasting. Krause [2] discussed and evaluated repeated decisions of individuals using a variant of the case-based decision theory, where individuals base their decisions on their own past experience and the experience of neighboring individuals, in looking at a range of scenarios to determine the successful outcome of a decision.

In summary, most abovementioned researches had traditional case-based reasoning techniques and related similarity measures mechanisms, but added the improved and developed novel concepts by themselves for stressing the importance of developed mechanisms by themselves. As mentioned above, the proposed system also developed related similarity measure mechanisms. That is to say, going through the Web service: CBR.InsTmpCascData provided by WIAS records the sequence of monitor data into the “TmpTable,” and then the agent 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 for easily completing a case-based reasoning and learning mechanism. By way of the abovementioned related system tableaux, the testing outcomes can still put greater stress on the specific similarity measure mechanism to be more suitable for application domains. Furthermore, to go through the properties of interface integration and real-time extension of Web services, the system performance is not only fast but corresponds to the research subject on the proposed cloud information services.

V. CONCLUSION AND DISCUSSION

The paper has proposed a cloud energy-saving and case-based reasoning information agent with Web service technique. Not only can it explore related technologies 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 energy-saving and a case-based reasoning information agent on the Internet. The preliminary system development, display and corresponding comparisons show that the research results are highly successful. Furthermore, the system evaluation shows around 40% of the data queries can be answered by the proposed system, which can effectively alleviate the overloading problem usually associated with a backend server, and its correct rate of data solutions is around 85.1%, which is a very efficient improvement in data query response. Truly extending related testing system tableaux for completing relevant interface and communication protocols of the proposed system, three-stage intelligent decision processing and final solutions on energy-saving processing to provide complete information services will be the focus of our future investigation.

ACKNOWLEDGEMENT

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, Taiwan, R.O.C., under Grants NSC-101-2221-E-129-001 and
REFERENCES


