Design and Implementation of Rule Based ORM Expert System

RAJENDRA PRASAD KURMACHALAM
RESEARCH SCHOLAR
OSMANIA UNIVERSITY, HYD

HARIKA YAMSANI M.Tech(CSE)
BALAJI INSTITUTE OF ENGINEERING & SCIENCE

Abstract
The development of the Design and implementation of any expert system, there are many problems should be considered and study carefully. In reality, human experts have common sense, deduction and analogical reasoning facilities. The proposed framework model for ORM expert system is based on the integration of two different knowledge representation formats (scheme) in the knowledge base. The proposed scheme is the mixing of the Rule-base and the Case-based forms using Blackboard in order to include the three facilities in one scheme. This Scheme will be facilitate applying more than one problem solving methods and search techniques in inference engine of the ORM expert system. The rule base and case base formats have been converted into tables. This paper presents the implementation of the proposed scheme as a Rule-Case-based ORM expert system. Also, presents illustrated examples of using the framework model together with the evaluation of the proposed framework model. The evaluation is based on the methodologies used for developing ORM expert system such as knowledge representations, searching technique, problem solving, and user interface design.

Keywords: ORM expert Systems, Problem Solving Methodologies, Search Techniques and Relational Database. Expert system, Design.

I. Introduction
The most important phase, in building knowledge based system and the expert systems, is the building of the knowledge base; this process is part of knowledge engineering which is an important field at present century. Usually, expert systems are designed and implemented for dedicated narrow and specific domain, while ORM expert system can be used for developing expert system in any domain, but ORM expert system are also governed by the format used for representation of the knowledge base [1]. The proposed scheme consists of the Rule-base and the Case-based formats using the Blackboard [2]. The scheme facilitates combination of forward and backward chaining reasoning, using many problem solving methodologies, and different searching techniques. The scheme makes the proposed Rule-Case-based ORM expert system more flexible, efficient, and more powerful for the development of the expert systems in future [3]. This view is based on the philosophy of human memory organization and utilizing for solving problems [4]-[5]. Therefore, the mixing of rule-base and the Case-based forms using Blackboard has not been used before for the ORM expert systems. Adapting the proposed scheme facilitates the common sense, deduction and analogical reasoning activities in the inference engine [6]. So this scheme will be used for the development of ORM expert system as Rule-Case-based ORM Expert System.

II. Rule based ORM Expert System
The ORM expert system is a complete development environment for building and maintaining knowledge-Based Applications and Expert Systems [7]. It provides a step-by-step methodology for a knowledge engineer that allows the domain experts themselves to be directly involved in structuring and encoding the knowledge. Most expert systems are developed via specialized software tools called ORM expert systems. These ORM come equipped with an inference mechanism.
(backward chaining, forward chaining, or both), and require knowledge to be entered according to a specified format, user interface, explanation facilities and editing facilities as seen in Fig. 1 [8].

![Architecture of Rule-Case-Based ORM Expert System](image)

Fig 1. Architecture of Rule-Case-Based ORM Expert System

Rule-based programming is one of the most commonly used techniques for developing expert systems. A rule-based expert system consists of a set of rules that can be repeatedly applied to a collection of facts. The following concepts are essential to rule-based systems: - Facts represent circumstances that describe a certain situation in the real world. Rules represent heuristics that define a set of actions to be executed in a given situation. There is a basic distinction between derivations and production rules. Derivation rules have the form if <condition> then <conclusion>, whereas production rules conclusion, in derivation rules, is abstract: it consists of deriving logical consequences from certain conditions. These logical consequences are simply asserted but not executed. An action, in production rules, is concrete: it consists of producing practical consequences from certain conditions. These practical consequences are concretely executed. ECA rules, namely rules like if <condition> then <action>, which are production rules. However, production rules can implement derivation rules by using a special action “assert”, which asserts knowledge. If the <action> part of a production rule is just a conclusion and not a function that performs actions, we can consider this production rule as a derivation rule. In both these cases, rules are composed of an if portion and a then portion. The if portion of a rule the left hand side (LHS), is called predicates or premises. The LHS consists of an expression, which can be a single expression (an individual fact that must be true for applying the rule) or a series of expressions (composite expression). In the literature of rule based languages, a single expression is usually called pattern. A composite expression consists of several single expressions connected together by using the conditional elements “and, or, not” in order to create complex rules. Usually, when several expressions are connected by the “and” conditional element, this element is omitted. In rule-based languages we have also the logical connectives “& | ~”, which are used to manipulate values inside a single fact [2]. User Interface is deal with all modules except Alarm System. Alarm system is background worker thread which will wait for an event to be occurred by time. Connection module is used to check network availability and overall network status of the installed system. If network connection is enabled, Network Status Gathering System module will set up the event to detect on specific time. Events will be configured in Network Status Gathering System. Administration tasks are to manage or set up an event to be an alarm. Which result of an alarm should notify to whom will be decided in this Administration task. Diagnostic is used to follow up the problems which encountered from the Alarm System’s result. Expert user (or) admin who handle the system will input configuration data to system. He will also make diagnosis with the system. System will present network information and alarm result to him. Network entities are network nodes which will be exist in the system network. System will check them according to user’s request or schedule tasks. Their status will be response to system to record or report. System will report alarm result to Other Technical User who defined/configured by admin. The overview flow chart of Network
Diagnosis Rule-Based Expert System is as shown in Fig. 3 The following procedures are processing steps for network diagnostic rule-based expert system. The tester has launched the application after finishes all require installations. When application has started, system login form can be seen. When tester typed password wrongly, it responses with an error message. After it has entered the valid password, main windows of the system is appeared. Before appear this main window, system load background worker thread of the alarm system according to configuration which had been entered by user last time. It means alarm system is enabled. If system is close, alarm system won't be existed in memory as alarm system is part of the system. When an error has occurred, user will be asked for making diagnostic When click the yes to make diagnostic according to diagnostic tree, first inference of respective fault will be shown and ask for user response. When it has reached end of the inference, diagnostic result, suggestion for fault and possibility will be shown according to inference tree.

III. Methods for Evaluation ORM Expert Systems
The evaluation is based on three other ORM expert systems together with the proposed Rule-Case-Based ORM Expert System (Al-NADA EXPSYS) using an evaluation methodology proposed by MD Salim, Alvaro Villavicencio, and Marc A. Timmerman [10]. The methodology uses two types of evaluation, either direct method or indirect method. The direct method generates data for "Satisfaction Level", which is a direct measure of the overall user satisfaction with the ORM. The indirect method is based on estimating the resources needed to write a program using general computer language to implement an Expert System. This type of calculation is based on finding answers to the following questions," If a user wrote a program in some computer language to solve this problem instead of using a ready-made Expert System ORM, how long would this program be, how many person-months would it take to write this program, and so on. In this paper the direct method is used and in the next subsection present the evaluation processes together with the results of the evaluation.

Evaluation Using Direct Method
This method is based on the factors which are related to the most important criteria's in the design and implementation of expert systems, which are; knowledge representation, search technique, problem solving method, and user interface. The method consists of many steps, these steps has been called instrument, and these steps are presented. The application of this method is presented.

IV. Hybrid Knowledge Base Scheme
The knowledge base represents the repository of knowledge for specific and narrow domain for the knowledge based system. So, the most important part of knowledge based system is the knowledge base and the power of any knowledge based system and Expert System inherently in the adequate and integration of knowledge representation forms used for the particular domain. In this sense, the most important phase, in building knowledge based system and the expert systems, is the building of the knowledge base; this process is part of knowledge engineering which is an important field at present century. Usually, expert systems are designed and implemented for dedicated narrow and specific domain, while sell expert system can be used for developing expert system in any domain, but ORM expert system are also governed by the format used for representation of the knowledge base. The proposed scheme consists of the Rule-base and the Case-based formats using the Blackboard. The scheme facilitates combination of forward and backward chaining reasoning, using
many problem solving methodologies, and different searching techniques. The scheme makes the proposed Rule-Case based ORM expert system more flexible, efficient, and more powerful for the development of the expert systems in future. This view is based on the philosophy of human memory organization and utilizing for solving problems. Usually human uses more than one form for knowledge representation in his long term memory in order to be more efficient for solving problems, also the knowledge of any domain can’t be in one format. In the literature survey, found that many publications have covered the development of knowledge-based systems into expert system, using case based reasoning in the areas of conceptual design, aircraft conflict resolution, military decision support systems, helpdesk operations, customer service management, legal systems, diagnosis, design, and planning [12]-[14]. It is seen that the applications of Case-Based Reasoning in developing knowledge-based systems and the expert systems have been widely adopted in various industries and other application areas [15]. Furthermore some applications have been incorporate other knowledge representation methods besides rule-based and case-based reasoning, such as neural networks and fuzzy logic [16]. Combination of forward chaining reasoning and backward chaining reasoning makes expert systems more flexible and efficient and also the use of more than one knowledge representation forms makes the expert system more powerful. Therefore, the mixing of rule base and the Case-based forms using Blackboard has not been used before for the ORM expert systems. The proposed scheme facilitates the common sense, deduction and analogical reasoning activities in the inference engine, because rule base provides the deduction, case base provides the analogical reasoning, and the blackboard provides the common sense, as seen in Fig. 2 [17].

**Fig2: Hybrid Knowledge Scheme**

**Rule mapping engine:** It’s an independent thread which access knowledge base directly and produce results to Rule-based knowledge system. Then the controller of the system fetches the knowledge according to rules formed from the front end. The controller system pushes the user data through the expert module, that data will formed as rules by expert system. Those rules mapped through rule mapping engine thread, this thread access the formed rules in dynamic data base. After rules mapped the mapped rules indexes of the data base given to expert system. And the controller can access data base through another thread. The results given by rule mapping engine and communication engine, ORM expert system produce a conclusion with explanation accurately. Rule mapping engine is useful to produce the results in short time and with accurate results.

The user interface simulates the communications facilities available to be used for interaction with the Rule-Case-Based ORM expert system. This means an information processing system of one of (vision, speech, hearing, touching, tasting) or specified protocol many be used to connect the ORM expert system to another computerized system. Usually the chosen
method or methods to interact with the ORM expert system will be based on format used for the representation of knowledge in the knowledge base. Since, the formats used in the proposed system will be a scheme of the integration of two formats, which are rule-base and casebase, so the user interface will be the appropriate communication facilities between the proposed Rule-Case- Based ORM expert system and the domain expert peoples. These facilities allow the user (the domain expert peoples) to create and update the knowledge-base during the development of the expert system. But if the proposed Rule-Case-Based ORM expert system connected to a computerized knowledge acquisition system then the interaction between two computer-based systems will be through the special protocols between them and should be appropriate with the proposed scheme for representation of the knowledge base.

V. Conclusion

The framework will be used for the development of ORM expert system as Rule-Case-based ORM Expert System. The framework uses both procedural and declarative knowledge representation formalisms through the application of relational data base. So the rule base and case base formats have been converted into tables. Adapting the proposed framework facilitates the common sense, deduction and analogical reasoning activities in the inference engine of the ORM expert system. The framework makes the proposed Rule-Case-based ORM expert system more flexible, efficient, and more powerful for the development of the expert systems in future. The proposed Rule-Case-Based ORM expert system can be connected to a computerized knowledge acquisition system then the interaction between two computer-based systems will be through the special protocols between them and should be appropriate with the proposed scheme for representation of the knowledge base. Also, incorporate other knowledge representation methods besides rule-based and case-based reasoning, such as neural networks and fuzzy logic.

References


Rajendra Prasad Research Scholar Osmania University, Hyderabad.

Harika Studying M.Tech in Balaji Institute of Engineering & Technology.