Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 4, April 2014, pg.739 – 743

RESEARCH ARTICLE

Optimization of Rule - Based Medication Delivery in Large scale Healthcare Framework

G.Manikandan[1], Dr. S.Prabakaran[2]

[1 2] Faculty of Engineering and Technology, SRM University, Kattankulathur, Chennai

[1] amtmanikandan@gmail.com

[2] prabhakaran.mani@gmail.com

Abstract— In large scale healthcare point of care areas, medication delivery to large number of patients simultaneously faces large number of barriers in terms of performance and scalability. Integrated decision support systems improve clinical performance and patient outcomes. Computerized programs have novel aspects that have to be considered but aspects such as technical problems/support and user interface issues acts as barriers. In terms of eliminating such barriers in large scale healthcare framework, optimization of rule-based expert system is highly necessary to attain high performance and scalability factors that ensure patient safety at Point of Care.

Keywords— Medication management; Expert System; Knowledge based

I. INTRODUCTION

A. Heuristic Approach

Problem solving is the process of finding a solution when the path leading to that solution is uncertain. Even though we are familiar with several problem-solving techniques, in the real world, sometimes many problems cannot be solved by a technique we are familiar with. Surprisingly, for some complicated problems, no straightforward solution technique is known at all. For these problems, heuristic solution techniques may be the only alternative. A heuristic can be simplified as a strategy that is powerful and general, but not absolutely guaranteed to provide best solutions. Heuristic methods are very problem specific. Previous experience and some general rules – often called rules of thumb could help find good heuristics easier. Of course, if the heuristic does fail, it is necessary for the problem solver to either pick another heuristic, or know that it is appropriate to give up. Choosing random solutions, adopting greedy approaches, evolving the basic heuristics for finding better heuristics are just some of the popular approaches used in heuristic problem solving Heuristic problem solving involves finding a set of rules, or a procedure, that finds satisfactory solutions to a specific problem. A good example is finding one's way through a better rule. To make the way toward the final goal, a step by- step movement is necessary to make necessary progress.

B. Physician Knowledge Base

A rule-based, expert system maintains a separation between its Knowledge-base and that part of the system that executes rules, often referred to as the expert system shell. The system shell is indifferent to the rules it executes. This is an important distinction, because it means that the expert system shell can be applied to many different problem domains with little or no change. It also means that adding or modifying rules to an expert system can effect changes in program behavior without affecting the controlling component, the system shell.

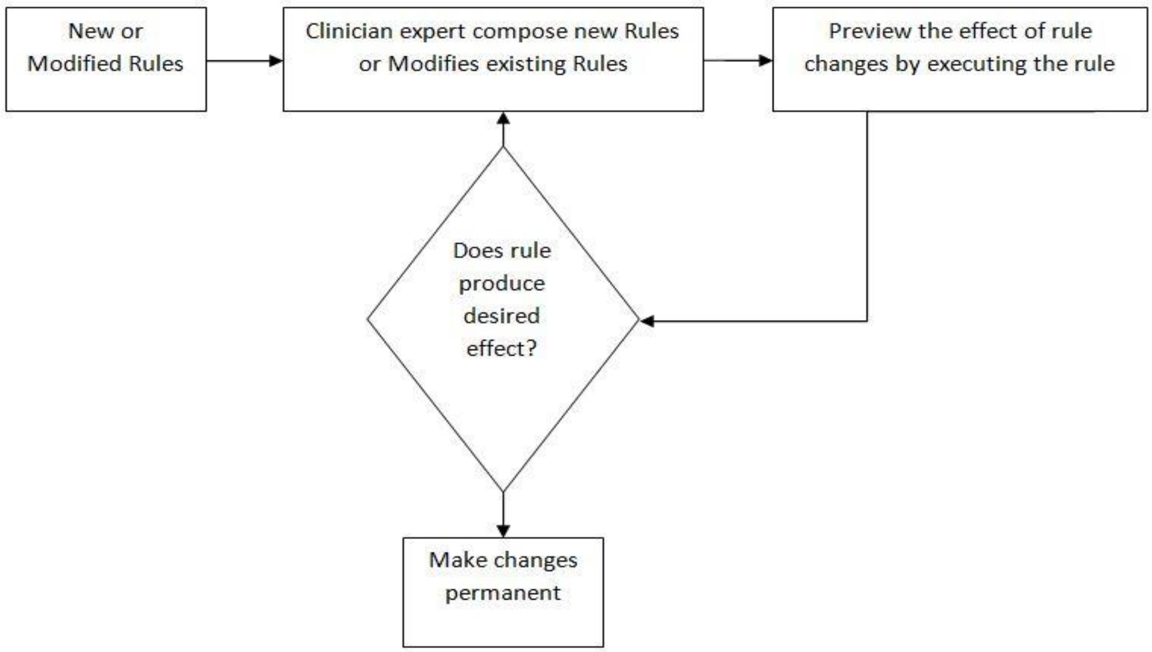


Fig.1 Adding RULES to the knowledge base

C. Client Interface

The Client Interface processes requests for service from system-users and from application layer components. Client Interface logic routes these requests to an appropriate shell program unit. For example, when a physician wishes to create or edit a rule, they use the Client Interface to dispatch the Knowledge-base Editor. Other service requests might schedule a rule, or a group of rules, for execution by the Rule Engine.

D. Rule Based Medication Delivery

A rule-based system is a set of "if-then" statements that uses a set of assertions, to which rules on how to act upon those assertions are created. In software development, rule-based systems can be used to create software that will provide an answer to a problem in place of a human expert. Rule based system includes,

- Rule base / Knowledge base
- An inference engine
- Match, Resolve, Act cycle
- Facts/Working memory
- User interface

This consists mainly of three things: facts, rules and an engine that acts on them. Rules represent knowledge and facts represent data. A rule-based system solves problems by applying rules on facts (i.e. matching facts with rules' *if* clauses). A rule consists of two parts: conditions (*if* clauses) and actions. The action part of a rule might assert new facts that fire other rules.

The idea of rule-based programming is to represent a domain expert's knowledge in a form called rules. Besides rules, another important ingredient in a rule-based system is facts. Say Mr.X is a physician. Here is the Mr.X's knowledge about a patient condition

1. If a patient had accident, give them Medication 1.
2. If a patient had accident, injury is severe and hence gives them Medication 1.

Representing this knowledge in a rule-based system as rules and facts,

E. Rules

Rule 1: If the patient had accident, give them Medication 1.

Rule 2: If the injury is severe, give them Medication 1.
Rule 3: If the patient had accident, injury is severe.

F. Facts

Fact 1: Patient had an accident.
Fact 2: Injury is severe.

The language used to express a rule is closely related to the language Physician use to describe problem solutions. When the Physician composes a rule using this language, he is, at the same time, creating a written record of problem knowledge, which can then be shared with others. The rule adds functionality or changes program behavior, and records essential information about the problem domain in a human-readable form. Knowledge captured and maintained by these systems ensures continuity of operations even as physician. Furthermore, changes to the Knowledge-base can be made easily by Physician without programmer intervention, thereby reducing the cost of software maintenance and helping to ensure that changes are made in the way they were intended. The shell portion includes software modules whose purpose it is to,

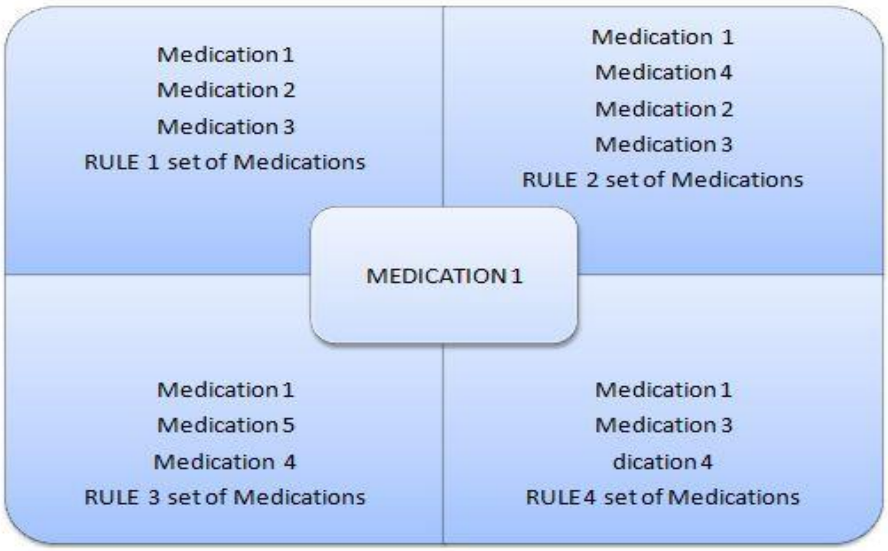
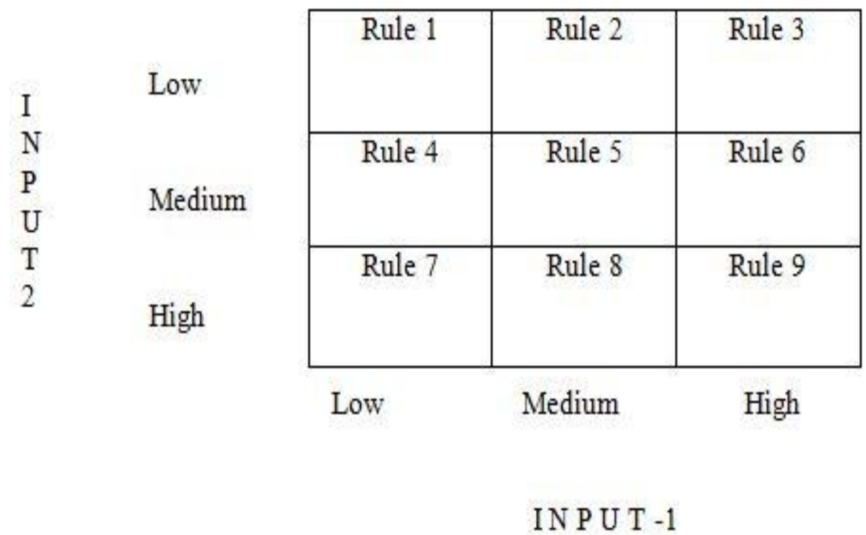


Fig.2 Medication Details in Rule sets

- Process requests for service from system users and application layer modules;
- Support the creation and modification of business rules by subject matter experts;
- Translate business rules, created by a subject matter experts, into machine-readable forms;
- Execute rules; and
- Provide low-level support to expert system components (e.g., retrieve metadata from and save metadata to knowledge base, build Abstract Syntax Trees during rule translation of business rules, etc.).

G. Illustration of Rule based abstraction

Consider in a hospital, a patient is admitted. Now we tell our expert system that a patient had accident by asserting fact number 1. Immediately the expert system tries to match the asserted fact with its three rules. Rules 1 and 3 are fired because the asserted fact satisfies their conditions (the *if* clause). When rule one is fired a more accurate term is activated, the system informs to give them Medication ABC as if it were a real physician.
Firing rule 3 asserts fact 2. And because fact 2 satisfies the conditions of rule 2, rule 2 is fired and the system says the injury is severe. All these chain reactions happen in sequence as the result of asserting fact number 1.



Example showing how the two-dimensional spaces are partitioned using three trapezoidal membership functions per input dimension. A simple if-then rule will appear as If input-1 is medium and input 2 is High, then rule R8 is fired.

H. Statistical Experimental model for Optimization

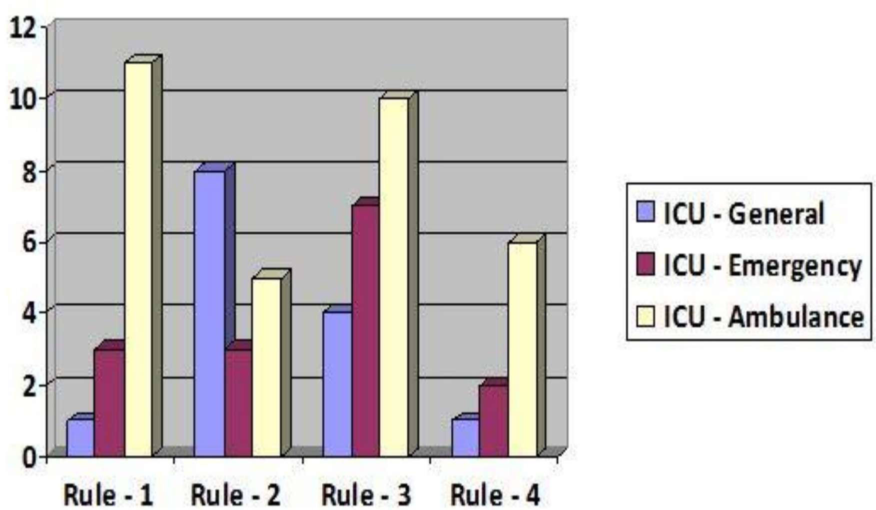


Fig.3 Medication delivered-All Rule sets

Optimization mechanism is extracting the exact rules used more often in the ICU point of care and utilizing it based on the need of emergency and this helps in immediate action of necessary steps to be taken to make patients life sustainable at a highly risk environment. Medications are selected from the rules executed frequently and it will reduce time in taking next emergency step of

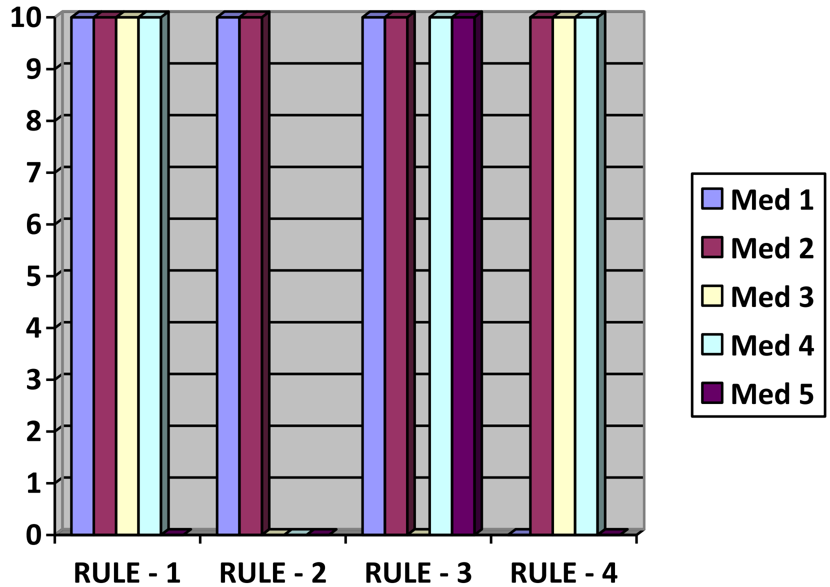


Fig.4 Medication 1 usage in all Rule sets

Here the rule sets we have Rule 1,Rule 2, Rule 3,Rule 4 and the sets of medications are medication 1, medication 2, medication 3, medication 4, and medication 5. In this, in all Rule Sets the optimization result has the rule set medication 1, that is needed more frequently. Thus we can extract medication 1 as the critical medication component that is highly necessary for all kinds of treatment and that can be kept readily available for other upcoming .

I. Summary

Rule-based expert systems have been applied in a vast number of application areas. An important advantage of the fuzzy expert system is that the knowledge is expressed as easy-to-understand linguistic rules. If we have data, the fuzzy expert system can be taught using neural network or other adaptation techniques. It is to be expected that the number of applications will grow considerably in the future.

REFERENCES

[1]Abraham, A. (2002) Intelligent Systems: Architectures and Perspectives, Recent Advances in Intelligent Paradigms and Applications, in "*Studies in Fuzziness and Soft Computing"*, Chapter 1, (eds A., Abraham, L., Jain and J., Kacprzyk), Springer Verlag,Germany (pp. 1–35).

[2]Ignizio, J.P. (1991) "*Introduction to Expert Systems: The Development and Implementation of Rule-Based Expert Systems"*,McGraw-Hill, Inc, USA

[3]Jang, J.S.R., Sun, C.T. and Mizutani, E. (1997) "*Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence"*, Prentice Hall Inc, USA.

[4]Paul Jones, Raoul Jetley and Jay Abraham"*A Formal Methods-based Verification Approach to Medical Device Software Analysis"*, February 2010.

[5]Raoul Jetley and Ben Chelf " *Diagnosing Medical Device Software Defects using Static Analysis. Medical Device & Diagnostic Industry"*, Volume 31, Number 5, Pages 72-83, May 2009.

[6]Raoul Jetley and Paul Anderson "*Using Static Analysis to Evaluate Software in Medical Devices. Embedded Systems Design"*, Volume 21, Number 4, Pages 40-44,April2008.

[7]Schneider, M., Langholz, G., Kandel, A. and Chew, G. (1996) *"Fuzzy Expert System Tools", John Wiley & Sons, USA.*