Reasoning with Cases in the CBR System: A Case Study for Applying *OOExpert* System

Romi Satria Wahono and Behrouz Homayoun Far
Department of Information and Computer Sciences, Saitama University
romi@cit.ics.saitama-u.ac.jp

Abstract:
Case-Based Reasoning (CBR) is a reasoning method that used in the intelligent systems to find useful and applicable old cases, and reuse them either directly or after adaptation. CBR enables information managers to increase efficiency and reduce cost by substantially automating processes such as diagnosis, scheduling and design. In this paper we introduce the essential characteristic of CBR and discuss the comparation of CBR with a range alternative decision support techniques. The problems and limitations of case-based reasoning and other reasoning methods are discussed. And a case study for implementing CBR to the *OOExpert* is also presented.

Keywords: Case-Based Reasoning, Intelligent System, *OOExpert*

1. Introduction

CBR is an artificial intelligence (AI) methodology that provides the foundations for a technology of intelligent systems. It has been used to develop many systems applied in a variety of domains, including manufacturing, design, law, medicine, and battle planning.

CBR is based on psychological theories of human cognition. It rests on the intuition that human expertise does not depend on rules or other formalized structures, but on experiences. Human experts differ from novices in their ability to relate problems to previous ones, to reason based on analogies between current and old problems, to use solutions from old experiences, and to recognize and avoid old errors and failures.

In this paper we introduce the essential characteristic of CBR and discuss the comparation of CBR with a range alternative decision support techniques. The problems and limitations of case-based reasoning and other reasoning methods are discussed. We also present the new hybrid architecture for object classes' identification in the *OOExpert* system, by integrating both CBR and Rule-Based Reasoning (RBR) paradigms.

2. CBR Versus Other Techniques

By comparing CBR with other computational techniques, including information retrieval (IR), rule-base reasoning (RBR) techniques, machine learning (ML), and neural networks (NN), we try to understand the strengths and weaknesses of CBR [7].

2.1. CBR Versus Information Retrieval

CBR and Information Retrieval (IR) have many features in common. IR can use techniques other than standard database queries, in particular when retrieving information from large textual information sources such as compact discs or the Internet. An increasingly popular retrieval technique is concept-based retrieval. This uses a thesaurus to find similes for words in a query to widen the scope of the query. Thus, a query such as "remedies for colds" would include word such as cure and treatment, which are synonymous with remedy, and fly and influenza, which are synonymous with cold.

CBR and IR both support flexible querying and both will retrieve a set of potentially relevant, but possibly inexact matches. However there are differences between the two techniques:

- IR methods are mainly focused upon retrieving text from large document sources, whereas CBR methods can deal with a wider range of data types.

- IR system do not tend to use background knowledge about the information being retrieved that CBR system can use.

Thus differs in that it tends to be used on richer information sources that IR techniques, which tend to be used only on textual databases. However, this distinction is being blurred, and CBR techniques are increasingly being used on large textual information sources.

2.2. CBR Versus Rule-Base Reasoning

Rule-based reasoning (RBR) breaks a problem down into a set of individual rules that each solves part of the problem. Rules are combined together to solve a whole problem. To create these rules, we have to know how to solve the problem, and this task can be extremely complex and time consuming.

CBR systems differ fundamentally in that to use them, we do not need to know how to solve a problem, only to recognize if we have solved a similar problem in the past. However, both RBR and CBR techniques are often used to solve similar problems such as fault diagnosis.

	RBR	CBR
Problem area	Narrow, well understood, strong domain theory, stable overtime	Wide, poorly understood, weak domain theory, dynamic overtime
Knowledge representation	Facts and IF-THEN rules	Cases
System provides	Answer	Precedents
Explanation by	Trace of fired rules	Precedent
System can learn	No, usually requires manual addition of new rules	Yes, by the case acquisition

Table 1: RBR Versus CBR

2.3. CBR Versus Machine Learning

In general, machine learning (ML) involves analyzing past cases to derive rules that applied to solve new problems. ML clearly separates the processes of learning rules and solving problems. CBR uses induction algorithms, to classify existing cases. The result of this process is an index tree that is used to match a new case against existing cases. Thus the distinction between learning and problem solving is less clearly separated.

Another important distinction is in the justification for decision or answer. ML problem solvers will justify a decision by quoting the rules that were induced from the training examples. A CBR system will use the retrieved cases as precedents to support a decision. This is an important distinction, since in general people understand and trust precedents but are less comfortable with abstract rules.

2.4. CBR Versus Neural Network

Superficially, there are some similarities between CBR and neural networks (NN). Both techniques rely on past cases with known outcomes to inform their decisions, but there the similarities end. NNs are good in domains where data cannot be represented symbolically. Conversely, CBR is less good with purely numeric data and much better with complex, structured symbolic data.

The group at Wolver Hampton University and the Heartlands Hospital also compared CBR (a nearest-neighbor system) to an NN (a multilayered perceptron trained by back propagation). The NN system could correctly advise on the dosage of warfarin in 79% of cases compared to 87% for the CBR system. The NN system therefore outperformed the linear discriminant analysis but was less accurate than CBR. However, we would expect NN technology to perform much better on data that was better suited to it.

The major disadvantage of NN technology compared with CBR is that an NN system functions as a “black box”. The answer given by an NN is a function of the weighted vectors of its neurons. No explanation or justification of any sort can be given by an NN. This is therefore even worse than RBR or ML systems, which can at least quote the rules used to justify a decision. For this reason, NNs are unsuitable in many application domains.

2.5. Summary of Technology Comparisons

Table 2 is a guide to when we should and should not use each of the technology discussed. The major limitations of CBR are that it may not handle large volumes of purely numeric data as well as statistical or neural network techniques, and that if complex adaptation is required to provide a precise or optimum answer.. CBR retrieves the most similar case and attempts to reuse the solution from that case. CBR does not provide precise, exact, or optimum solutions.

Technology Type	When To Use	When Not To Use
Database	Well-structured, standardized data and simple precise queries possible	Complex, poorly structured data and fuzzy queries required
Information Retrieval	Large volume of textual data	Nontextual complex data types, background knowledge available
Rule-Based Reasoning	Well-understood, stable, narrow problem area and justification by rule-trace acceptable	Poorly understood problem area that constantly changes
Machine Learning	Generalizable rules are required from a large training set and justification by rule-trace is acceptable	Rules are not required, and justification by rule-trace is unacceptable
Neural Network	Noisy numerical data for pattern recognition or signal processing	Complex symbolic data or when a justification is required
Case-Based Reasoning	Poorly understood problem area with complex structured data that changes slowly with time and justification required	When case data is no t available, or if complex adaptation is required or if an exact optimum answer is required

Table 2: Technology Comparisons

3. Implementing CBR in the OOExpert

In our research project, we are developing an expert system that aims to help designers while designing object-oriented software by automating the difficulties and ill-defined tasks in object-oriented design process, including identification of object classes, object classes relationship, object classes attributes etc. This system was named *OOExpert*.

In the *OOExpert* system, RBR and CBR have been combined. The complementary properties of CBR and RBR can be advantageously combined to solve some problems to which only one technique fails to provide a satisfactory solution. Generally the combination involves CBR systems using RBR for support. RBR and CBR are often used together, where the use of rules is supplemented with the use of cases that determine the scope of the rules. CBR processing can be augmented with RBR when general domain knowledge is required.

Figure 1 shows the architecture of the object classes' identification in *OOExpert* system by using RBR and CBR integration approach.

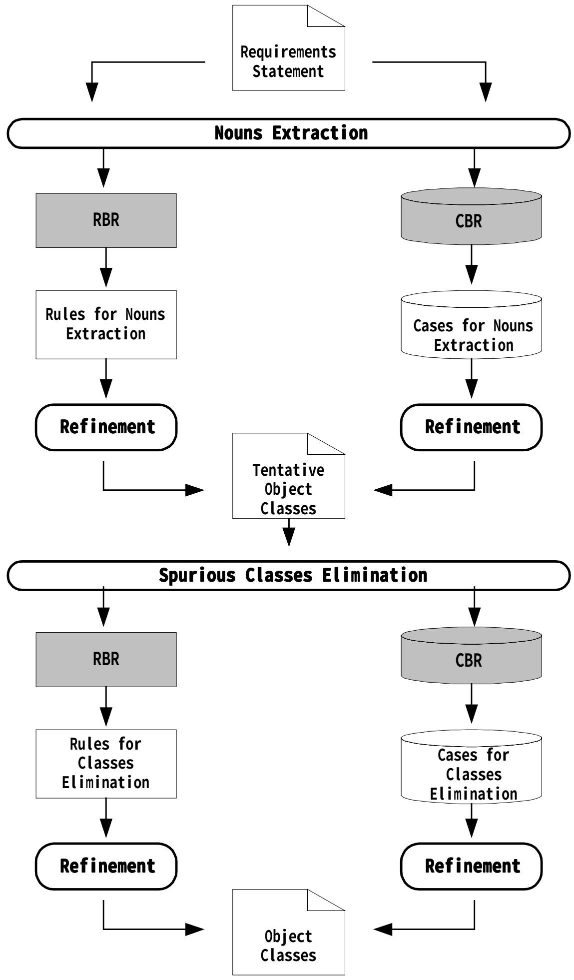

Figure 1: Architecture of the Object Classes' Identification in OOExpert

The first step constructing an object model is to identify relevant object classes from the application domain. Objects include physical entities, such as houses, employees, and machines, as well as concepts, such as trajectories, seating assignments, and payment schedules. All classes must make sense in the application domain. As shown in Figure 2, begin by listing candidate object classes found in the written description of the problem. Classes often respond to nouns. Then *OOExpert*'s *reasoning engine* will process this nouns extraction request by using rules from rule-base and cases (experiences) from case-base. As a result we have tentative object classes.

The next step is to eliminate spurious classes. In RBR, the system will discard unnecessary and incorrect classes according to the following criteria: redundant classes, irrelevant classes, vague classes, attributes, operations, roles, and classes that point at implementation constructs.

In other hand, CBR is based on psychological theories of human cognition. We collect design rules from human experts, and store/index them in the case-base. It rests on the intuition that human expertise does not depend on rules or other formalized structures, but on experiences. Human experts differ from novices in their ability to relate problems to previous ones, to reason based on analogies between current and old problems, to use solutions from old experiences, and to recognize and avoid old errors and failures. Using cases from case-base, we can get another solutions of identifying object classes, from experiences of human experts.

Using this integration approach, RBR and CBR have been combined in *OOExpert* system to engender performance improvements and to solve problems to which single technique fails to provide a satisfactory solution.

4. Conclusions

In this paper we introduced the essential characteristic of CBR and discuss the comparation of CBR with a range alternative decision support techniques. The problems and limitations of case-based reasoning and other reasoning methods are discussed. And we presented the new hybrid architecture by integrating both CBR and Rule-Based Reasoning (RBR) paradigms for object classes' identification in the *OOExpert*.

Using this integration approach, RBR and CBR have been combined in *OOExpert* system to engender performance improvements and to solve problems to which single technique fails to provide a satisfactory solution.

5. References

[1] Romi Satria Wahono and Behrouz Homayoun Far, "Distributed Expert System Architecture for Automatic Object-Oriented Software Design," *Proceedings of The Third Workshop on Electro-Communication and Information*, Japan, March, 1999.

[2] Romi Satria Wahono and Behrouz Homayoun Far, "OOExpert: Distributed Expert System for Automatic Object-Oriented Software Design," *Proceedings of the 13th Annual Conference of JSAI*, Japan, June 1999.

[3] Ian M. Holland and Karl J. Lieberherr, "Object-Oriented Design.", *ACM Computing Surveys*, Vol. 28, No. 1, March 1996.

[4] Nick Cercone, Aijun An, and Christine Chan, "Rule-Induction and Case-Based Reasoning: Hybrid Architectures Appear Advantageous", *IEEE Transactions on Knowledge Engineering and Data Engineering*, Vol. 11, No.1, January/February 1999.

[5] 加藤貞行, "オブジェクト識別についての一考察とその効果",*情報処理学会研究報告*, Vol.90, No.100,1998.

[6] Sycara E.P., "Finding Creative Solutions in Adversarial Impasses", *Proceedings of the Ninth Annual Conference o the Cognitive Science Society*, 1987.

[7] Ian Watson, "Applying Case-Based Reasoning: Technique for Enterprise Systems", *Morgan Kaufmann*, 1997.

[8] Friedrich Gebhardt, Angi Vob, Wolfgang Grather, Barbara Schmidt-Belz, "Reasoning With Complex Cases", *Kluwer Academic Publisher*, 1997

Biography of Author

Romi Satria Wahono, Was born in Madiun-Indonesia on October 2nd 1974, Received B.Eng. in Information and Computer Sciences, in 1999, from Saitama University. He is currently a researcher at the Center for Scientific Documentation and Information (PDII-LIPI), and a M.Eng. candidate at the Department of Information and Computer Sciences, Saitama University. The research fields of his interests are Distributed Artificial Intelligence, Multi Agent Systems, Reasoning and Object-Orientation. He is a member of the Association for Computing Machinery (ACM), IEEE Computer Society, Japanese Society for Artificial Intelligence (JSAI), and Indonesian Society on Electrical, Electronics, Communication and Information (IECI).

Behrouz Homayoun Far, Received BSc. and MSc. degrees in Electronic Engineering in 1983 and 1986, respectively, from Tehran University, Iran. He has received his Ph.D. degree from Chiba University - Japan, in 1990. He is currently an Associate Professor at the Department of Information and Computer Sciences, Saitama University - Japan. The research fields of his interest are qualitative reasoning, automatic programming and distributed AI. Dr. Far is a member of the ACM, IEEE Computer society, Japanese Society for Artificial Intelligence, IEICE and Information Processing Society of Japan.