

An Integrated Knowledge Adaption Framework for Case-based Reasoning Systems

Ning Lu, Jie Lu, Guangquan Zhang

Faculty of Engineering and Information Technology
University of Technology Sydney, PO Box 123, Broadway, NSW 2007, Australia
Email: {philiplu, jielu, zhangg}@it.uts.edu.au

Abstract. The development of effective knowledge adaption techniques is one of the promising solutions to improve the performance of case-based reasoning (CBR) systems. Case-base maintenance becomes a powerful method to refine knowledge in CBR systems. This paper proposes an integrated knowledge adaption framework for CBR systems which contains a meta database component and a maintenance strategies component. The meta database component can help track changes of interested concepts and therefore enable a CBR system to signal a need for maintenance or to invoke adaption on its own. The maintenance strategies component can perform cross-container maintenance operations in a CBR system. This paper also illustrates how the proposed integrated knowledge adaption framework assists decision makers to build dynamic prediction and decision capabilities.

Keywords: knowledge adaption, case-based reasoning, case-base maintenance, machine learning, adaptive decision support

1 Introduction

Organizational decision makers are expected to carry out tasks and solve problems effectively and efficiently by learning previous knowledge. Machine learning provides automatic techniques to learn past knowledge, observations, and various experience in order to make accurate predictions and support decision-making. A main concern in machine learning is the design and development of learning algorithms that can continuously acquire and refine knowledge, such as rules and patterns, from data, and allow computers to improve their performance over time.

One of the current challenges facing machine learning algorithms, like CART, ID3, C4.5 and IFN [1], is to handle the dynamic changing situations. In many machine learning-based systems, the distributions or patterns underlying the data are changing over time rather than remaining stable, which is also known as concept drift [2]. As a result, when a learning algorithm considers all the past training examples or makes an assumption that training data is a random sample drawn from a stationary distribution, the induced patterns may not relevant to the new data [3, 4]. This raises an urgent need of an effective learner which is able to track such changes and to quickly adapt to them [2].

Case-based reasoning (CBR), as one of the most popular machine learning technique, shows the process of solving new problems based on the solutions of similar past problems [5]. The prediction and reasoning accuracy is mainly depended on the quality of cases and other knowledge. In order to perform as well as possible despite changing circumstances, a CBR system must be able to signal the need for maintenance or to invoke maintenance strategies as needed, that is to develop an adaptive CBR system [6]. This leads to a new active research area, case-based maintenance (CBM), which has been seen as a new phase of CBR. Literature reports several CBM policies to improve efficiency, reserve competency and maintain consistency of a CBR system. The most creative point of the CBM technique is that it allows a CBR system to update itself dynamically by taking information from problem solving process. Obviously, this ability can greatly improve the performance of CBR in applications. We therefore can use CBM techniques to support the establishment of adaptive CBR systems.

The success of long-term CBR applications in dynamic environment depends critically on the accuracy and refining of the contents of case bases and other knowledge containers — vocabulary knowledge, retrieval knowledge (similarity measure), and adaption knowledge (solution transformation) [7]. To provide maintenance support for a case base, Smyth and Keane [8] suggested a competence-preserving case deletion policy which introduces two important competence properties, coverage and reachability. Delany & Cunningham [9] enhanced this competence model by adding a liability property and proposed a conservative redundancy reduction policy. In the meantime, Zhu & Yang [10] developed a case-addition policy, which repeatedly selects cases from an original case-base and then adds them into an empty case-base until reaches a certain size limit. Moreover, some researches of CBM extended beyond maintaining case base alone and addressed to all the other knowledge containers of a CBR system. Kim & Han [11] described a GA-based case representation which can search for the near-optimal form of representation through discretization; Craw, Jarmulak & Rowe [12] developed a method to optimize CBR retrieval and Hanney & Keane [13] presented an inductive learning algorithm to extract adaption rules from the cases in the case-base which is very useful in CBM. All these proposed methods can maintain case base and other knowledge, but they mainly focus on single knowledge container only. In real world applications, a completed CBR system contains a set of knowledge containers and must be able to adapt knowledge within all these containers as a whole. Heister and Wilke [14] identified this problem and suggested a repair phase to keep consistency among knowledge containers as well as a framework to support the new phase. However, their framework is weak in tracking changes and lack of discussion of relationships between adaption knowledge and other knowledge in a CBR system. To handle these two issues, we propose an integrated knowledge adaption framework, which has a meta database component to help track changes of concepts thus enable the CBR system to signal a need for maintenance and a maintenance strategies component for cross-container adaption operations.

The paper is organized as follow. In Section 2, we briefly explain related researches and developments in CBR and CBM. In Section 3, we present an integrated knowledge adaption framework for CBR systems. Comparing with others, the advantages of the framework is it gives detailed approaches to enhance the

communication between a CBM/CBR system and its users and to restore all knowledge containers dynamically. Section 4 gives an example to illustrate how the framework design commits a cross-container adaption for CBR. Conclusions and further studies are discussed in Section 5.

2 Related Works

This section will first review the concepts and methods of case-based reasoning. It will then introduce the new development in CBR research by considering CBM.

2.1 Case-based Reasoning

Case-based reasoning offers a powerful learning ability to use past experiences as a basis for dealing with new problems, and facilitates the knowledge acquisition process by avoiding the time required to elicit solutions from experts. It is represented by a four-step (4Rs) cycle: *retrieve*, *reuse*, *revise* and *retain* [5]. A new case defined by an initial problem description is used to *retrieve* a case from the base of previous cases. “The retrieved case is combined with the new case - through *reuse* - into a solved case, i.e. a proposed solution to the initial problem. Through the *revise* process this solution is tested for success, e.g. by being applied to the real world environment or evaluated by a teacher, and repaired if failed. During *retain*, useful experience is retained for future reuse, and the case base is updated by a new learned case, or by modification of some existing cases.”[5]

2.2 Extended Case-based Reasoning

The success of a CBR system is mainly dependent on the suitability of knowledge and the correctness of reasoning. Therefore, certain “harmful” knowledge may actually degrade system performance [8]. To cope with all issues that arise when CBR is used in real world applications and especially in rapid changing environments, some additional phases such as maintenance phase has been accepted to extend original CBR cycle [15]. As a typical development, Reinartz [16] extended the standard four-step CBR cycle [5] by two additional steps: *review* and *restore*. The *review* step covers tasks to judge and monitor the current state of a CBR system and its knowledge containers, whereas the *restore* step invokes mechanisms to change the system and its knowledge to improve the performance. Figure 1 shows the six-step CBR cycle.

2.3 Case-base Maintenance

The case-base maintenance (CBM) refers to the process of refining a CBR system to improve the system’s performance, which implements policies of revising the contents of a CBR system in order to facilitate future reasoning [17]. To guide the

general study of maintenance in a CBR system, Wilson and Leake [17] provided a common framework for describing CBM policies. They categorized CBM approaches in terms of CBM policies that determine when and how a CBR system performs CBM. But they only gave a very general framework, which didn't give enough design and explanation about how operations of different knowledge containers affect each other and how to integrate maintenance operations of different containers into a CBR system.

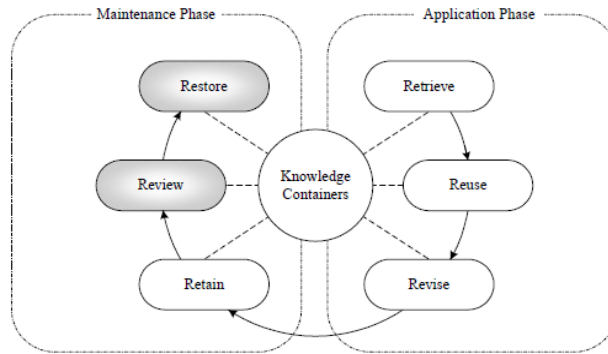


Fig. 1. The Six-Step CBR Cycle [16]

To address this issue, Heister and Wilke [14] presented a two-phase approach. They defined maintenance operations as modifications of a system made by the users in an intended manner and repair operations as pre-defined scripts to be performed after a maintenance operation has happened to keep system consistent. They further developed a maintenance architecture to support these two operations during changes of the system. However, their framework is not able to track changes of concepts, and thus doesn't facilitate the decisions of when, where and how to make changes.

3 An Integrated Knowledge Adaption Framework for CBR

To provide better communications with the decision makers and facilitate their decisions with regards to knowledge maintenance issue, we proposed an integrated knowledge adaption framework for CBR systems. The framework is able to track changes of concepts, thus signal a need for maintenance to decision makers or even perform a series of operations on its own. It also supports changes of multiple knowledge containers and cross-container operations. Coincide with the extended six-step of CBR cycle [16], the framework consists of two parts, application phase (the right half) and maintenance phase (the left part), as shown in Figure 2. This paper mainly explains the *meta database* component (corresponding to the *review* step in Figure 2) and *maintenance strategies* component (corresponding to the *restore* step in Figure 2).

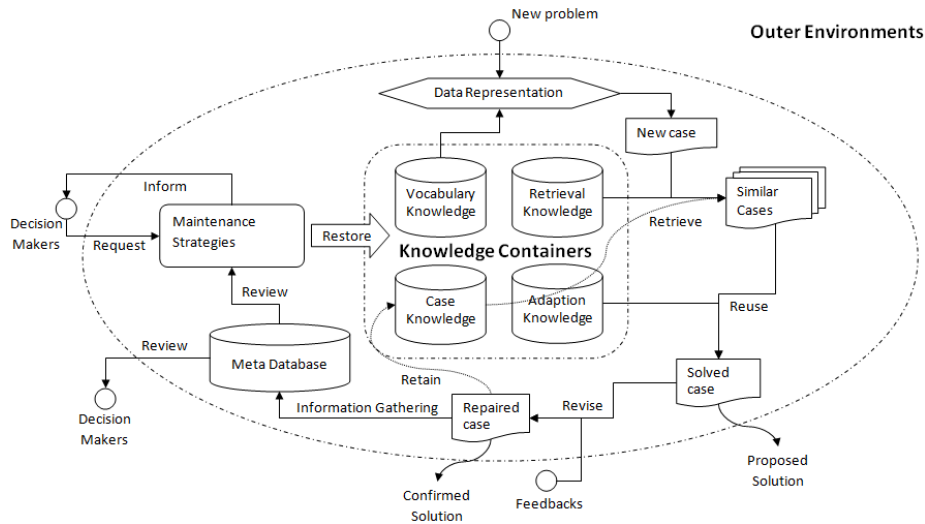


Fig. 2. Integrated knowledge Adaption Framework for Case-based Reasoning System

3.1 Meta Database

Followed by the general study of maintenance policies [17], we defined a meta database which stores the information gathered about individual cases, about the case base as a whole or in part, and about the overall processing behaviour of the CBR system. The information gathered at individual case level includes the number of times that a case has been reused successfully or the times it has failed, the histories of case retention or reuse and the current measurement of the coverage set and reachability set for a case [18]. The information gathered at case base level includes cases in groups which support the generalized adaption knowledge or trained similarity measures and the size of current case base. The information gathered for processing behaviour includes the average retrieval time of the system, a set of problems that the system cannot solve successfully or cost highly for solving. By reviewing or monitoring the information gathered in it, one could assess the current performance of a system, thus perform or trigger a certain set of maintenance operations when desired.

3.2 Maintenance Strategies

The result of data analysis serves as input for determining whether CBM is necessary and where to restore. Thus, the maintenance strategies component takes *meta data* as input and restores the knowledge in a CBR system. While a system may inform users that certain maintenance operations are needed, such as removing noise cases, refining similarity measures, users may also request certain operations in desired, such as adding new adaption rules, adding new similarity measures. As changes of one knowledge container may also affect others, the proposed or requested

maintenance operations will be followed by certain sets of repair operations to prohibit inconsistency as shown in Figure 3. The results of these operations will be evaluated by the performance improvement of a CBR system or the decision makers, and the system might be reverted in case of failure. We will give an example in the next section to illustrate how the changes will be made.

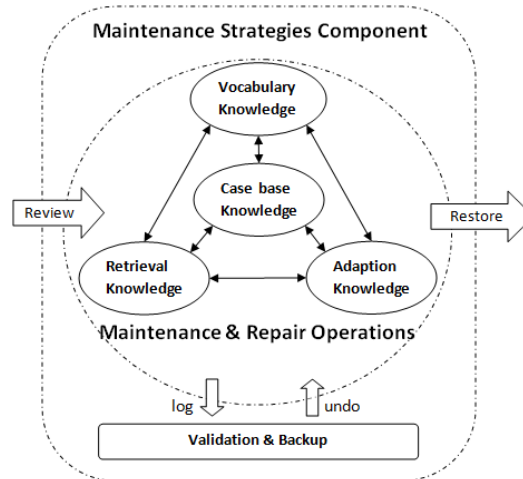


Fig. 3. Maintenance Strategies Component

4 A Case Study

Suppose there is a CBR system designed for a telecommunication company, which helps the marketing department to determine strategies on how to attract more customers. In the past, to attract the youth, several cases indicated that a low charge of SMS was a good champion strategy (depicted as circles in Figure 4). Recently, as iphone becomes a new hot spot and the functionality of playing multimedia is more attractive, the original strategy does not work well in the competition. To solve this problem, new solutions are worked out and retained as new cases in the case base (stars in Figure 4), such as offering a good price for iphone. However, when marketing decision-makers consider both scenarios, they will get confused, as the two strategies (expressed by two groups of cases) are conflict with each other. Thus case base maintenance will be needed.

As shown in Figure 4, by reviewing *meta data*, the decision makers are able to recognize increasing inconsistency for solving the problem of how to attract the youth and a decreasing reliability of circle cases. Thus removing of such old fashioned cases will certainly improve the performance of the CBR system. This proposed system can also automatically identify these 'old' cases and inform/perform a case deletion operation through suitable review and evaluations.

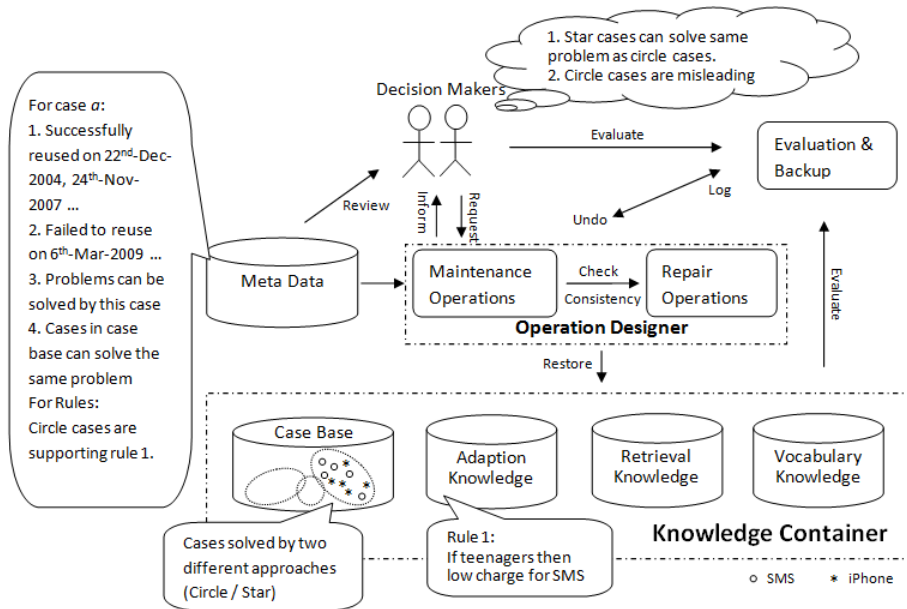


Fig. 4. An Example of Case-base Maintenance under the Integrated Knowledge Adaption Framework

In addition, changes in a case base may affect other knowledge in the CBR system, especially when the original retrieval and adaption knowledge is obtained based on the case base. Take the same example, as the result of the removal of circle cases in the case base, the support of generalized adaption knowledge — Rule 1 (Figure 4), becomes weak. Thus certain repair operations may need to be performed to keep system consistent. For example, Rule 1 will be replaced by new rules generated by the remaining cases in the case-base.

5 Conclusions and Future works

Traditional machine learning methods are facing challenges when applied in dynamic changing environments. For CBR systems, CBM has already been accepted as an essential phase. In this paper, we have proposed an integrated knowledge adaption framework which integrates all knowledge containers together and supports cross-container operations. Especially, this framework adopted a meta database component which gathers information from three different aspects and helps to determinate when and where to restore a CBR system.

For future work, noise cases always mislead the adaption process and can hardly be distinguished with newly concepts. Maintenance strategies under different environments still need further studies and investigations. Second, maintenance strategies themselves could also be seen as knowledge in a CBR system. Studies on how to assess and refine this kind knowledge will be another task.

Acknowledgment. The work presented in this paper was supported by Australian Research Council (ARC) under Discovery Project DP0880739.

References

1. Maimon, O. and M. Last, *Knowledge Discovery and Data Mining - The Info-Fuzzy Network (IFN) Methodology* 2000, Boston: Kluwer Academic.
2. Widmer, G. and M. Kubat, *Learning in the Presence of Concept Drift and Hidden Contexts*. Machine Learning, 1996. **23**(1): p. 69-101.
3. Geoff, H., S. Laurie, and D. Pedro, *Mining time-changing data streams*, in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. 2001, ACM: San Francisco, California.
4. Cohen, L., et al., *Info-fuzzy algorithms for mining dynamic data streams*. Applied Soft Computing, 2008. **8**(4): p. 1283-1294.
5. Aamodt, A. and E. Plaza, *Case-based reasoning: Foundational issues, methodological variations, and system approaches*. AI Communications, 1994. **7**(1): p. 39-59.
6. Leake, D. and D. Wilson, *When Experience is Wrong: Examining CBR for Changing Tasks and Environments*, in *Case-Based Reasoning Research and Development*. 1999. p. 720-720.
7. Richter, M.M., ed. *Introduction*. Case-Based Reasoning Technology, From Foundations to Applications, ed. M. Lenz, et al. 1998, Springer-Verlag: Berlin. 1-15.
8. Smyth, B. and M.T. Keane. *Remembering To Forget: A Competence-Preserving Case Deletion Policy for Case-Based Reasoning Systems*. in *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*. 1995. San Francisco: Morgan Kaufmann.
9. Delany, S.J. and P. Cunningham, *An Analysis of Case-Base Editing in a Spam Filtering System*, in *Advances in Case-Based Reasoning*. 2004. p. 128-141.
10. Zhu, J. and Q. Yang. *Remembering to Add: Competence-preserving Case-Addition Policies for Case-Base Maintenance*. in *In Proceedings of the International Joint Conference in Artificial Intelligence (IJCAI)*. 1999: Morgan Kaufmann.
11. Kim, K.-j. and I. Han, *Maintaining case-based reasoning systems using a genetic algorithms approach*. Expert Systems with Applications, 2001. **21**(3): p. 139-145.
12. Craw, S., J. Jarmulak, and R. Rowe, *Maintaining Retrieval Knowledge in a Case-Based Reasoning System*. Computational Intelligence, 2001. **17**(2): p. 346.
13. Hanney, K. and M. Keane, *Learning adaptation rules from a case-base*, in *Advances in Case-Based Reasoning*. 1996. p. 179-192.
14. Heister, F. and W. Wilke, *An Architecture for Maintaining Case-Based Reasoning Systems*, in *Advances in Case-Based Reasoning*. 1998. p. 221.
15. Göker, M.H. and T. Roth-Berghofer, *The development and utilization of the case-based help-desk support system HOMER*. Engineering Applications of Artificial Intelligence, 1999. **12**(6): p. 665-680.
16. Reinartz, T., I. Iglezakis, and T. Roth-Berghofer, *Review and Restore for Case-Base Maintenance*. Computational Intelligence, 2001. **17**(2): p. 214.
17. Wilson, D.C. and D.B. Leake, *Maintaining Case-Based Reasoners: Dimensions and Directions*. Computational Intelligence, 2001. **17**(2): p. 196.
18. McKenna, E. and B. Smyth, *Competence-Guided Case-Base Editing Techniques*, in *Advances in Case-Based Reasoning*. 2000. p. 235-257.