Mini-Tutorial presented at the TRIO/ITRC Research Retreat, Queen's University, May 10-12, 1994

Representation and Management Issues for Case-Based Reasoning Systems

Igor Jurišica[1]
juris@cs.utoronto.ca
Department of Computer Science
University of Toronto
Toronto, Ontario M5S 1A4, Canada

Abstract

The objective of the tutorial is to present the mechanisms and techniques underlying case-based reasoning. A special emphasis will be on CBR's potential as an alternative to current knowledge-based systems. We will explore both theoretical and practical issues, identify advantages of CBR compared to other reasoning methods, and characterize application domains where CBR seems to be most promising.

[1]Supported by the University of Toronto Open Doctoral Fellowship and a Research Assistantship from the Department of Computer Science

A case-based reasoning system (CBRS) can be seen as a special type of a knowledge-based system. Case-based systems deal with cases or episodes, which represent instances of concepts (or their generic form). They are special in the sense of different knowledge they use and consequently different representation and management requirements they have. Case-based reasoning also shares similar features with similarity-based reasoning, instance-based reasoning, analogy-based reasoning, and reinforcement learning. The major processes shared by the reasoner that does case-based reasoning are as follows:

1. **Case storage** - representing all important features of the episode (experience) in a case base. In general, a case consists of a problem description and an outcome. Representation of cases can be flat or can be hierarchically organized.

2. **Case retrieval** - retrieving relevant cases from the case base. In general, a similarity assessment is used to locate the most relevant cases which are then used to create a solution.

Case-based reasoning (CBR) is usually viewed as a process of remembering and recalling old experience. The decision making is based on comparisons between the new situation and the old experience; thus, the quality of reasoning depends on the quality and quantity of cases in a case base. With increased number of unique cases, problem-solving capabilities of CBR systems improve while the effectiveness of the system (at least in terms of efficiency) may decrease.

Case-Based Reasoning Systems

Even though case-based reasoning systems share the major processes (storage and retrieval), they differ in decision making methods. Based on this classification, the following systems are usually recognized:

1. **Interpretative systems** compare new situation to recalled experience. The interpretation is used when the problem is not well understood and when there is a need to criticize a solution. Cases are used to provide justification for solutions, evaluation of solutions when no clear-cut methods are available, and interpretation of situations when definitions of the situation's boundaries are not well-defined. A natural application for interpretative systems is the American legal system: relevant cases form an argument for or against a case. There is no need to modify the past case, the problem is solved by finding and presenting the relevant cases.

2. **Problem-solving systems** derive solutions to a new problems by adapting a solution for the solved past experience. Case-based reasoning system is considered to be a problem solver if:

   - it can learn from mistakes, which will prevent the system from repeating the same mistake twice;
   - it will keep trying to find better solution, even though one is already available;
   - it will not consider available results immediately - it will first modify them to fit the current needs.

   A problem-solving CBRS is applied to problems, where simple presentation of past cases does not suffice for finding a solution (e.g., planning, diagnosis, design). In such a situation, a case needs to be modified to fit a new problem: some parts may be deleted because they

are not needed for the new problem, some parts may be added since they are missing in the old case, and some parts may be changed in a specific way. This process may also rely on generalizations, specializations, or on analogical problem-solving.

The performance of case-based reasoning system directly depends on the quality and quantity of its encoded knowledge: on the one hand, if there is not enough knowledge, the problem-solving capabilities are limited; on the other hand, the system can become useless because of unmanageable amount of knowledge.

The representation and management issues for case-based reasoning are challenging because many applications require large case bases with space- and time-efficient processing. Even though the representation and management of cases can be handled similarly as in knowledge-based systems, there are particular differences which need to be considered, namely:

- Cases are suitable for capturing problem-solving processes by representing episodes of these processes, and thus are suitable not only in easily formalizable areas (as are traditional knowledge-based systems) but in less formalizable areas as well. Because knowledge acquisition is supported by learning from past experience, CBRSs are able to supplement weak domain models.

- A case can represent a complete solution (plan, design) which can be retrieved, adapted, and reused later, eliminating the need for decomposition and recomposition.

- In contrast to earlier AI systems, case-based system use specific past experience in solving problems (however, accumulated cases can be processed and generalized later.).

Case Representation, Classification, Retrieval, and Adaptation

There still is a lack of consensus as to exactly what information should be represented in a case. In general, a case is an arbitrary set of features, used for describing a particular concept. Considering its form, a case can be a simple concept or a connected set of subcases; it can be a specific instance or a generalization. In addition to the representations presented so far, more emphasis should be put towards characterizing useful contents of cases, such as context, task, and feedback information. Formalization of case representation is the goal of research as well. This will also allow for easier case knowledge interchange and case base reuse by various systems, which is non-existent so far.

Similarity plays a central role in theories of human problem solving and thus is important for the research in artificial intelligence. Even though there are different approaches to similarity, the underlying idea is to classify cases according to some features which allow us to use these cases in similar situations.

Retrieval of cases from memory can be characterized as remembering of relevant past experience. Retrieval consists of the following subtasks:

1. **Recall previous cases** - the goal is to retrieve cases relevant to the current problem, e.g., retrieve similar cases. Cases are found according to classification which is created according to features in the current case.

2. **Select the best subset** - only the most promising case (or cases) retrieved in the initial step are selected. For some application only one case is required, for other, small set of relevant cases is important (for adaptation and/or justification).

There are only a few examples where the old solution can be reused without any change: either if the case base is sufficiently large, or if cases are represented with different granularities (fine-grain cases comprise a coarse-grain cases; thus, the solution can be composed using fine-grain cases). More often, one must modify either a solution or strategy to get a solution for the current problem. The process of modifying the old example to fit a new problem is called adaptation. At the beginning, the system has to decide what should be adapted in the ballpark solution. Later, the adaptation rules are applied to take the differences between the retrieved case and the input case into account.

Case-Based Reasoning System Applications

Application areas of case-based systems include case-based design, case-based planning, case-based diagnosis, and case-based explanation.

In *case-based design*, problems are defined as a set of constraints; the problem solver is required to provide a concrete artifact that solves the constraint problem.

Case-based planning and case-based design systems are quite similar to each other. Both types of systems are useful for the problems that are hard to solve in one chunk, but cannot be easily decomposed, and solved separately, because of the strong interaction between individual pieces. Here, individual steps imply different constraints which have to be satisfied.

In *case-based diagnosis*, a solver is given a set of symptoms and is asked to explain them. On the one hand, diagnosis is similar to the classification problem in cases when only a small number of possible explanations exist. On the other hand, if the space of possible explanations is significant, one can view diagnosis as the problem of creating an explanation. Similarly to the comparison of design and planning case-based systems, case-based diagnosis system can rely on planning and thus can be classified as a planning system. However, there are also diagnosis systems which are model-based. In these cases, categorization is easier since the distinctive features are easily perceivable.

Explanation is introduced to the CBR terminology in the connection with the plan or solution failure. The system may try to construct an explanation of the reason for the failure in order that the plan or solution may be repaired and the case base modified to avoid such a failure in the future.

Evaluating Case-Based Systems

System evaluation can include not only performance measure issues but also verification and validation. Verification is performed to ensure consistency, completeness, and correctness of the system. Even though most of the systems rely on the ensured consistency there are also systems which do not require consistency checking. Validation ensures correctness of the final system with respect to the user needs and requirements. On the one hand, verification of the CBR system is virtually non-existent.

Conclusion

Case-based reasoning paradigm is a promising approach to solving complex problems. A variety of research prototypes, industrial applications, and commercial CBR shells is available.

A comprehensive introduction to the area of case-based reasoning is presented in Janet Kolodner's book: *Case-Based Reasoning.* Morgan Kaufmann, San Mateo, CA. 1993, discussing both theoretical and practical achievements.