Learning to Improve Case Adaptation by Introspective Reasoning and CBR*

David B. Leake, Andrew Kinley, and David Wilson

Computer Science Department
Lindley Hall 215, Indiana University
Bloomington, IN 47405, U.S.A.

Abstract. In current CBR systems, case adaptation is usually performed by rule-based methods that use task-specific rules hand-coded by the system developer. The ability to define those rules depends on knowledge of the task and domain that may not be available *a priori*, presenting a serious impediment to endowing CBR systems with the needed adaptation knowledge. This paper describes ongoing research on a method to address this problem by acquiring adaptation knowledge from experience. The method uses reasoning from scratch, based on introspective reasoning about the requirements for successful adaptation, to build up a library of *adaptation cases* that are stored for future reuse. We describe the tenets of the approach and the types of knowledge it requires. We sketch initial computer implementation, lessons learned, and open questions for further study.

1 Introduction

Case-based reasoning (CBR) systems solve new problems by *retrieving* prior solutions of similar previous problems and performing *case adaptation* (also called *case modification*) to fit the retrieved cases to the new situation. Although much progress has been made in methods for case retrieval, both the American and European CBR communities have identified case adaptation as a particularly challenging open problem for the field (e.g., [1, 18]). The problem is so acute that the most effective current strategy for building CBR applications is to bypass adaptation entirely, building advisory systems that provide cases to human users who perform the adaptation themselves (e.g., [2, 14]). However, despite the practical benefits of retrieval-only advisory systems, successful use of advisory systems may require considerable user expertise. Consequently, automatic case adaptation is important from a practical perspective, not only to enable CBR systems to perform autonomously but to enable them to aid naive users. Likewise, as we discuss in [19], increased understanding of the case adaptation

* This work was supported by the National Science Foundation under Grant No. IRI-9409348. To appear in *Proceedings of the First International Conference on Case-Based Reasoning*, Sesimbra, Portugal, 1995.

process and the knowledge required is also important from a cognitive modeling perspective, as a step towards understanding how humans adapt cases when they reason from prior episodes.

This paper describes research based on characterizing case adaptation knowledge by decomposing it into two parts: (1) a small set of abstract structural transformations (e.g., [5, 9]), and (2) memory search strategies for finding the information needed to apply those transformations. This framework forms the basis of an approach to adaptation in which new adaptation problems are solved by first selecting a transformation indexed under the type of problem motivating adaptation, and then performing introspective reasoning about how to strategically search memory for the information needed to apply the transformation [17]. Not only does this approach provide increased flexibility in finding needed information, but it serves as a foundation for learning to improve adaptation performance from experience: A trace of this process can be stored as an *adaptation case* and used in future case-based reasoning about the adaptation process itself. Thus the approach is aimed at providing both the flexibility to deal with novel case adaptation problems and adaptation abilities that improve with experience.

We begin by discussing the significance of the case adaptation problem for CBR and the tenets of our approach. We then summarize an initial implementation that applies our approach to learning case adaptation for case-based planning (e.g., [8]) in the disaster response planning domain. We close by highlighting lessons learned and related research on case adaptation and memory search.

2 Acquiring Case Adaptation Knowledge

Coding effective adaptation rules can require extensive knowledge of the CBR system's task, its domain, and the contents of its memory. Unfortunately, this knowledge may not be available *a priori*. Thus in defining case adaptation rules, developers face the same problem of knowledge acquisition in imperfectly-understood domains that often impedes the development of rule-based systems in other contexts. In many of those contexts, the knowledge acquisition problem has been significantly ameliorated by the use of case-based reasoning. Consequently, it is natural to consider applying CBR to the case adaptation process itself, replacing pre-defined adaptation rules with adaptation cases that reflect prior adaptation experience [3, 17, 29].

An important question is the source of the needed library of adaptation cases. We propose a method that starts with a library of domain-independent adaptation rules, using them to solve novel adaptation problems. The results of applying those rules to specific adaptation problems are stored as adaptation cases to be re-used by case-based reasoning. The following sections first discuss the rule-based process and then the use of adaptation cases.

2.1 Adaptation = Transformations + Memory Search

Case adaptation knowledge is often characterized in either of two ways. The first is with abstract rules, such as the rule *add a step to remove harmful side-*

effect for case-based planning [8]. Such rules are applicable to a broad class of plan adaptation problems, but give no guidance about *how* to find the specific knowledge needed to apply them (e.g., to find the right step to add in order to mitigate a given side-effect). For example, if the planning task is to generate X-ray treatment plans, and the retrieved plan administers the minimum X-ray dose required to destroy a tumor, but also has the bad side-effect of exposing the spinal cord to excessive radiation, deciding which step to add in order to remove the bad side-effect may require considerable domain knowledge.

The second way to characterize adaptations is by relying on adaptation rules that include the required specific knowledge. For example, in the radiation treatment planning domain, the general rule *add a step to remove harmful side-effect* can be replaced by specific rules such as *add the step "rotate radiation sources" to remove harmful side-effect "excess radiation"* [4].

Both these approaches exhibit the classic operationality/generality tradeoff from explanation-based learning (e.g., [28]). Abstract rules have generality: a small set of transformations appears sufficient to characterize a wide range of adaptations [5, 15]. However, abstract rules are difficult to apply. Specific rules, on the other hand, are easy to apply but have limited generality. In addition, defining such rules is difficult because of the specific knowledge that they require.

Kass [11] proposes one way to address the operationality/generality tradeoff. His approach uses hand-coded *adaptation strategies* that combine general transformations with domain-independent memory search strategies for finding the domain-specific information needed to apply the strategies. Our approach to adaptation builds on this idea in treating adaptation knowledge as a combination of knowledge about general transformations and about memory search. However, instead of relying on hand-coded memory search strategies, our model builds memory search strategies as needed. When presented with a novel adaptation problem, it performs a planning process that reasons introspectively to determine the information required to solve the particular adaptation problem and to decide which memory search strategies to use to find that information. This process guides the search for information needed to perform the adaptation.

2.2 From Rule-Based Adaptation to CBR

After an adaptation problem has been solved by reasoning from scratch, a natural question is how to learn from that reasoning. Initially, it appears that explanation-based generalization (EBG) (e.g., [22]). would be the appropriate learning method, because it allows forming operational new generalizations: The memory search plan that found the needed information could be generalized and stored. However, one of the conclusions of our research is that using EBG to learn memory search rules is not practical [17]. For EBG to apply successfully to memory search rules, those memory search rules must provide a complete and correct theory of the contents and organization of memory. Unfortunately, the contents and organization of a specific memory are highly idiosyncratic [13, 27] and thus hard to characterize precisely. Consequently, a chain of memory search rules that finds desired information in one instance is not guaranteed to apply to other problems that appear to be within the scope of those same rules: explanation-based generalization may not yield reliable results.

In contrast, using case-based reasoning as the learning method for adaptation knowledge makes it possible for learned knowledge to reflect the idiosyncrasies of the memory's organization and its contents; unlike abstract adaptation rules, cases that package particular adaptation episodes encapsulate the system's experience on specific adaptation and memory search problems and reflect the system's specific task, domain, and memory organization. Consequently, we are applying CBR to learning adaptation cases. Thus our model acquires not only a library of problem-solving cases, but also a library of cases representing episodes of case adaptation. The following section discusses our computer model of the entire adaptation process, including both case adaptation from scratch in response to novel adaptation problems and case-based adaptation to re-use the results of previous adaptation episodes.

3 DIAL

The task domain for our research is *disaster response planning* for natural and man-made disasters. Examples of such disasters include earthquakes, chemical spills, and "sick building syndrome," in which occupants of a building fall victim to problems caused by low air quality inside a building. Studies of human disaster response planning support that case-based reasoning plays an important role in response planning by human disaster planners [26].

Our computer model, the case-based planner DIAL,[2] starts with a library of domain cases—disaster response plans from previous disasters—and general (domain-independent) rules about case adaptation and memory search. Like other case-base planners, it learns new plans by storing the results of its planning process. However, the central focus of our research is not on the case-based planning process *per se,* but on learning to improve case adaptation.

When DIAL successfully adapts a response plan to a new situation, it stores not only the problem solving episode, but also two types of adaptation knowledge for use in similar future adaptation problems: *memory search cases* encapsulating information about the steps in the memory search process, and *adaptation cases* encapsulating information about the adaptation problem as a whole, the memory search cases used to solve it, and the solution to the adaptation problem.

The entire DIAL system includes a schema-based story understander (that receives its input in a conceptual representation), a response plan retriever and instantiator, a simple evaluator for candidate response plans, and an adaptation component to adapt plans when problems are found. The case-based planning framework is based in a straightforward way on previous case-based planners (e.g., CHEF [8]). Consequently, this paper will only discuss the adaptation component.

[2] For **D**isaster response with **I**ntrospective **A**daptation **L**earning.

DIAL's adaptation component receives two inputs: an instantiated disaster response plan and a description of a problem in the response plan requiring adaptation. To illustrate, one of the examples processed by DIAL involves the following story: *At Beaver Meadow Elementary School in Concord, New Hampshire, students have been complaining of symptoms like unusual fatigue, eye irritation, respiratory problems, and allergic reactions from being inside the building.* When DIAL processes this story, a straightforward schema-based understanding process identifies the problem as an air quality problem. DIAL then attempts to retrieve and apply a response plan for a similar disaster. The response plan retrieved is the plan for the following factory air quality problem: *A & D Manufacturing in Bangor, Maine, has recently come under pressure from workers and union-representatives to correct perceived environmental problems in the building. Workers have been affected by severe respiratory problems, headaches, fatigue, and dizziness.* (These episodes are based on case studies from the *INvironment* newsletter for indoor air quality consultants.)

The response plan for A & D Manufacturing involves notifying the workers' union. DIAL's evaluator determines that the notification step does not apply to the current situation, because of a conflict with normative type restrictions on union members: elementary school students do not belong to unions. (The evaluation and problem characterization process is similar to that described in Leake [16]). Consequently, the response plan must be adapted to apply to the students. DIAL's adaptation component receives two inputs describing this situation: the response plan for the A & D Manufacturing problem, applied to the new situation, and a description of the problem to repair by adaptation: that trying to notify the students' union is not reasonable, because students do not belong to unions. After a description of the general processing done in response to adaptation problems, we will discuss how it applies to this example.

Given inputs describing a candidate response plan and a problem to be adapted, the process performed by DIAL's adaptation component is as follows:

1. **Case-based adaptation:** DIAL first attempts to retrieve an adaptation case that applied successfully to a similar previous problem. If retrieval is successful, that case is re-applied and processing continues with step 3.
2. **Rule-based adaptation:** When no relevant prior case is retrieved, DIAL selects a transformation associated with the type of problem that is being adapted (e.g., role/filler mismatches, such as the mismatch between unions and students, are associated with substitution transformations: a mismatch can be repaired by replacing the role being filled or how the given role is filled). Given the transformation, the program generates a *knowledge goal* [23] for the information needed to apply the transformation. E.g., for substitutions of role-fillers, the knowledge goal is to find an object that satisfies all the case's constraints on the object being replaced.

   The knowledge goal is then passed to a planning component that uses introspective reasoning about alternative memory search strategies [17, 20] to find the information needed. This search process generates a memory search plan whose operators include both an initial set of memory search strategies and *memory search cases* stored after solving previous adaptation problems.

3. **Plan evaluation:** The adapted response plan is evaluated by a simple evaluator that checks the compatibility of the current plan with explicit constraints from the response plan. A human user performs backup evaluation. If the new response plan is not acceptable, other adaptations are tried.
4. **Storage:** When adaptation is successful, the resulting response plan, adaptation case, and memory search plan are stored for future use.

The following subsections elaborate on the representation of knowledge goals, the memory search process, the adaptation case representation, and the examples currently processed.

Representing knowledge goals: In order to use our framework to guide rule-based case adaptation, a CBR system must be able to reason about how to find the information that it needs in order to apply a given transformation to a particular response plan. To do this reasoning, it must first have an explicit representation of the sought-after information. In DIAL, these needs are represented by explicit *knowledge goals* [23]. Previous study of knowledge goals has developed a two-part representation combining a *concept specification* [23] providing a template to match with candidate information and a description of how the information, once found, should be used.

To satisfy the requirements of memory search, however, we have found that the representation must include some additional components. First, as is reflected implicitly in the retrieval mechanisms of many CBR systems, the goals of memory search must often be described in terms of the available alternatives in memory (e.g., searching for the matching problem whose solution appears easiest to adapt, compared to other alternatives), rather than described by simply matching a template. Consequently, DIAL's knowledge goal representation also includes a *comparative specification* describing how to choose between multiple alternatives that satisfy the concept specification. Also, DIAL's knowledge goal representation includes information on the amount of search effort allowed for satisfying the knowledge goal (measured in terms of the number of primitive memory operations that may be applied during memory search).

The memory search process: During DIAL's initial rule-based adaptation process, it finds the information needed to apply adaptation transformations by an introspective reasoning process that implements memory search as a form of planning, using operators that describe actions within its internal, or "mental" world, rather than within the external world [10]. Using a planning process facilitates flexible re-combination of memory search knowledge. By decoupling memory search knowledge from specific adaptation rules, memory search knowledge can be applied to any problem for which it appears relevant.

Two types of memory search knowledge are provided to the system. First, the system is provided with *knowledge goal transformation rules*, similar to Kolodner's [13] query transformation rules, that reformulate the questions posed to memory. For example, one strategy for retrieving an instance of an event is to search for contexts in which it would have been likely to play a role. Second, the system is provided with a suite of domain-independent *memory search strategies* that depend on "weak methods" of memory search (e.g., ascending and descending abstraction hierarchies to find related nodes). DIAL currently includes six of these strategies. All strategies are defined in terms of a substrate of seven primitive memory access operations (e.g., to extract the "parent" of a node).

The results of the memory search process are filtered by constraints from the particular adaptation problem. The result is a relatively unguided initial search for information, but traces of this process are saved as *memory search cases* and made accessible for use during future memory search. These cases provide more precise guidance for memory search in similar future situations. In this model, cases are acquired solely by reasoning from scratch, which may require considerable processing effort. However, as will be discussed in a later section, we have also begun to investigate how this view of adaptation can be used to facilitate interactive acquisition of adaptation knowledge.

DIAL's memory search mechanism uses a reactive planning framework, inspired by the RAPS system [7], to interleave planning with execution and respond to problems during memory search (e.g., that needed intermediate information cannot be found). In this process, DIAL's rule-based adapter accepts a knowledge goal and chooses a strategy or stored memory search case indexed by the knowledge goal. In the course of processing, a strategy may transform the current knowledge goal or may generate sub-knowledge-goals, also to be satisfied by the planning process. Throughout the memory search process, the adapter maintains a reasoning trace of the operators it applies. That trace is packaged with the search result, as a memory search case, and stored for future use.

Representing and organizing cases learned from adaptation episodes: DIAL's *memory search cases* package the initial knowledge goal, a trace of knowledge-goal transformations and other memory search operations involved in the search process, a record of the search outcome (failure or success), the cost of the search in terms of primitive memory operations performed, and the information found. Memory search cases are indexed under the knowledge goals that they satisfy, and can suggest search operations to attempt in the future; they also have the potential to be used to warn of previous search failures. Memory search cases are accessible to the knowledge planning process for memory search, augmenting the initial library of built-in operators. For future searches, successful search cases that match the largest subset of the current knowledge goals are re-used. When the result of the stored search case does not satisfy current constraints, the search is continued by local search.

DIAL also packages *adaptation cases*, which include both the transformation used for the adaptation and pointers to memory search cases used to search for information to apply the transformation. These provide more specific guidance about how to adapt cases to repair particular types of problems.

Examples processed: DIAL's initial case library currently contains two disaster response plans, a response plan for the previously-described air quality disaster at A & D Manufacturing and a response plan for an industrial chemical spill. The system has been tested on four different stories exercising different parts of its adaptation mechanisms. The first concerns the indoor air quality problem at Beaver Meadow school, for which DIAL retrieves the A & D disaster response plan. (Like the stories processed, stored response plans are based on episodes from the *INvironment* newsletter.) The A & D disaster response plan includes many steps applicable to the new situation, providing the basis for a response to the school air quality problem. However, as previously described, one of the steps in the response plan for the air quality problem at A & D manufacturing does not apply: notifying the union of the victims. Because schoolchildren do not have unions, the notification step of the previous response plan must be adapted to apply to the schoolchildren. Many adaptations are possible, but a common suggestion from human readers is that the step involving notifying the union should be adapted into a step notifying the children's parents.

When DIAL is run on this example, no adaptations have yet been learned, so the program uses its rule-based adaptation process to perform the adaptation. It first selects a substitution transformation. (In DIAL, candidate transformations for repairing problems in retrieved cases are indexed directly under categories of problem types. For a description of possible problem types, see Leake, 1992.) In this case, the "role/filler mismatch" problem of the schoolchildren belonging to a union may be resolved by either of two substitutions: substituting a new filler (notifying someone else's union) or substituting a different concept in which the children play a similar role (notifying another group relevant to the children). To determine appropriate substitutions, the system must hypothesize the factors that were important in the relationship between workers and their union in the A & D manufacturing problem. Possible constraints can be obtained by examining alternative "views" of the relationship between the union and the workers in the original episode [31], based on the relationships represented in the system's memory. In DIAL's memory, one view of union membership involves the member *being represented*, suggesting searching for representatives of the children. This search yields "parents" as one possibility. (Other possibilities, like "student government" are also hypothesized but rejected during evaluation.) By storing the successful choices according to internal and external feedback, the system builds up information beyond the information in its initial world model about which adaptations to favor for particular adaptation problems.

A second example involves an air quality problem on a military base. The A & D manufacturing episode is the most similar in memory, but again the step of notifying the union fails to apply, this time because soldiers do not have unions. DIAL retrieves the previously-learned adaptation but finds that it too fails to apply: Notifying the soldier's parents is rejected by the user. Consequently, it applies a very simple adaptation to the adaptation case, discarding the final step in the memory search plan from the adaptation case and adding local search. In particular, it preserves that the *representation* relationship was important in the previous situation, and searches for representatives of soldiers. Using this guidance, it searches memory for representatives of soldiers and finds "commanding officers" as a possible group to notify.

Two additional examples involve another disaster at a school, to which the Beaver Meadow school response plan is reapplied in a straightforward way without adaptation, and the story of a chemical spill episode at a school. The chemical spill example illustrates the importance of learning new *adaptations* during CBR, instead of only learning new *cases* as traditionally done in CBR systems. For the chemical spill example, DIAL retrieves the previous chemical spill example as the most similar *case*, which is reasonable in light of the shared steps involved in cleaning up chemical spills—the response plan learned from the Beaver Meadow air quality problem is not the most similar response plan. However, the *adaptation* learned from processing the Beaver Meadow story is still useful: DIAL uses the adaptation learned from the Beaver Meadow school example to adapt the response to the previous chemical spill (which also involves notifying the workers' union) by substituting the students' parents. This demonstrates the value of decoupling case learning from adaptation learning: learning both new adaptation cases and new problem-solving cases increases the effectiveness of a CBR system in responding to new problems.

4 Lessons Learned and Open Issues

The conclusions drawn from the project to date include a number of points discussed in the previous sections: the usefulness of decomposing adaptation knowledge into two semi-independent parts, abstract transformations and memory search knowledge; the appropriateness of CBR; rather than explanation-based learning, as the mechanism for learning the needed memory search information; the need for a richer notion of knowledge goals than in previous research; and the usefulness of a reactive model of memory search planning in order to use incremental results of the search to guide further decisions.

Learning new strategies for adapting cases also has interesting ramifications for similarity assessment. In current CBR systems, similarity assessment is generally based on fixed criteria. However, as a CBR system learns how to adapt cases to deal with new types of problems, the similarity metric should be adjusted to reflect that (thanks to the adaptation learning), those problems are no longer as great an impediment to applying the case. Consequently, one area for further study is how best to make the similarity assessment process reflect the changing state of system adaptation knowledge.

We are now addressing a number of open questions. One of these is the level of granularity to be used for memory search cases. At present, memory search cases package entire memory search plans, but is possible that making subparts of the search plans available, as in Redmond's [24] *snippets,* would be beneficial.

Another question being studied is the effectiveness of the planful memory search process. To give an indication of the value of the knowledge planning framework for memory search, the current examples have been processed both using the planful process and using the simple *local search* strategy used by a number of CBR systems to find substitutions [15]. In this comparison, the knowledge planning method resulted in an order of magnitude savings in the number of primitive memory operations performed. This improvement is encouraging, although at this point it cannot be taken too seriously because of the limited set of examples used. Likewise, not enough examples are yet implemented to have reliable data on the tradeoffs between memory search by knowledge planning and CBR. We are now extending the system with the aim of performing additional tests. In particular, an important tradeoff to investigate is the *utility problem* [21] for learned adaptation knowledge: the danger that processing overhead due to the proliferation of adaptation cases and memory search cases will counterbalance the benefits of the additional guidance that they provide.

A final question involves the potential to apply this view of case adaptation to alternative methods for acquiring case adaptation knowledge. DIAL models the transition from adaptation by using unguided general rules to adaptation by using specific adaptation cases, by storing results of successful rule-based adaptation. With its method, the initial rule-based adaptation phase may be quite expensive. An alternative method for acquiring adaptation cases is to use its view of adaptation—as transformations plus memory search—as a basis for an interface to facilitate direct acquisition of adaptation cases from a human user. Such an interface could enable a user to suggest transformations and search strategies from a vocabulary of alternatives. We have begun to investigate this approach, both for its own potential and as a means of more rapidly acquiring a set of adaptation cases to test and refine DIAL's case-based adaptation process.

5 Relationship to Other Approaches

Memory search: Although many sophisticated memory search schemes have been developed in CBR research, they are normally driven by opaque procedures, rather than being accessible to explicit reasoning and learning. Our research follows an alternative course, developing explicit models of the memory search process to increase the flexibility and effectiveness of memory search, in the spirit of [13, 25], and to make it accessible to learning, as in [6, 12].

Case adaptation: Some previous systems are able to learn knowledge useful for guiding adaptation. For example, although CHEF [8] has a static library of domain-independent plan repair strategies, it augments that library with learned *ingredient critics* that suggest adaptations appropriate to particular ingredients. Likewise, PERSUADER [29] uses a combination of heuristics and case-based reasoning to guide adaptation, searching memory for similar prior adaptations to apply. In these systems, however, the adaptation information learned is quite domain and task specific, while memory search cases have more flexibility. The use of CBR for case adaptation has also been advocated by Berger [3], in the context of storing and re-using an expert's adaptations. An alternative approach to the case adaptation problem is to use derivational analogy, deriving a new solution by re-applying a prior solution process to new circumstances, rather than directly adapting the old solution itself [30].

6 Conclusions

Automatic case adaptation is necessary to enable CBR systems to function autonomously and to serve naive as well as expert users. However, knowledge acquisition problems for the rule-based adaptation methods used in many CBR systems have proven a serious impediment to developing CBR applications that perform their own adaptation.

We have described a framework for characterizing adaptation knowledge in terms of transformations and information search, have discussed how that framework is being used as the basis for a model of automatic learning of case adaptation knowledge, and have sketched an initial implementation of that model.

The model combines reasoning from scratch and case-based reasoning to build up expertise at case adaptation. The aim of this approach is to enable CBR systems to make the transition from adaptation guided by general rules (which may be unreliable and expensive to apply) to adaptation guided by adaptation cases that reflect specific case adaptation experience. Thus our method is a way for CBR systems to learn to become more effective at applying their existing cases to new situations.

References

1. Dean Allemang. Review of the first European workshop on case based reasoning EWCBR-93. *Case-Based Reasoning Newsletter*, 2(3), 1993. Electronic newsletter, special interest group AK-CBR, German Society for Computer Science.
2. R. Barletta. A hybrid indexing and retrieval strategy for advisory CBR systems built with ReMind. In *Proceedings of the Second European Workshop on Case-Based Reasoning*, pages 49–58, Chantilly, France, 1994.
3. J. Berger. Using past repair episodes. Unpublished manuscript, August 1995.
4. J. Berger and K. Hammond. ROENTGEN: a memory-based approach to radiation therapy treatment. In R. Bareiss, editor, *Proceedings of the Case-Based Reasoning Workshop*, pages 203–214, San Mateo, 1991. DARPA, Morgan Kaufmann, Inc.
5. J. Carbonell. Learning by analogy: Formulating and generalizing plans from past experience. In R. Michalski, J. Carbonell, and T. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*. Morgan Kaufmann, San Mateo, CA, 1983.
6. M. Cox. Machines that forget: Learning from retrieval failure of mis-indexed explanations. In *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*, pages 225–230, Atlanta, GA, 1994.
7. R.J. Firby. *Adaptive Execution in Complex Dynamic Worlds*. PhD thesis, Yale University, 1989. Computer Science Department TR 672.
8. K. Hammond. *Case-Based Planning: Viewing Planning as a Memory Task*. Academic Press, San Diego, 1989.
9. T. Hinrichs. *Problem Solving in Open Worlds: A Case Study in Design*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1992.
10. L. Hunter. Planning to learn. In *Proceedings of the Twelfth Annual Conference of the Cognitive Science Society*, pages 261–268, Cambridge, MA, July 1990. Cognitive Science Society.

11. A. Kass. Tweaker: Adapting old explanations to new situations. In R.C. Schank, C. Riesbeck, and A. Kass, editors, *Inside Case-Based Explanation*, chapter 8, pages 263–295. Lawrence Erlbaum Associates, 1994.
12. A. Kennedy. Using a domain-independent introspection mechanism to improve memory search. In *Proceedings of the 1995 AAAI Spring Symposium on Representing Mental States and Mechanisms*, Stanford, CA, March 1995. AAAI.
13. J. Kolodner. *Retrieval and Organizational Strategies in Conceptual Memory*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1984.
14. J. Kolodner. Improving human decision making through case-based decision aiding. *The AI Magazine*, 12(2):52–68, Summer 1991.
15. J. Kolodner. *Case-Based Reasoning*. Morgan Kaufmann, San Mateo, CA, 1993.
16. D. Leake. *Evaluating Explanations: A Content Theory*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1992.
17. D. Leake. Towards a computer model of memory search strategy learning. In *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*, pages 549–554, Atlanta, GA, 1994.
18. D. Leake. Workshop report: The AAAI-93 workshop on case-based reasoning. *The AI Magazine*, 15(1):63–64, 1994.
19. D. Leake. Combining rules and cases to learn case adaptation. In *Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society*, Pittsburgh, PA, 1995. In press.
20. D. Leake. Representing self-knowledge for introspection about memory search. In *Proceedings of the 1995 AAAI Spring Symposium on Representing Mental States and Mechanisms*, pages 84–88, Stanford, CA, March 1995. AAAI.
21. S. Minton. *Learning Search Control Knowledge: An Explanation-Based Approach*. Kluwer Academic Publishers, Boston, 1988.
22. T. Mitchell, R. Keller, and S. Kedar-Cabelli. Explanation-based generalization: A unifying view. *Machine Learning*, 1(1):47–80, 1986.
23. Ashwin Ram. AQUA: Asking questions and understanding answers. In *Proceedings of the Sixth Annual National Conference on Artificial Intelligence*, pages 312–316, Seattle, WA, July 1987. Morgan Kaufmann Publishers, Inc.
24. M. Redmond. *Learning by Observing and Understanding Expert Problem Solving*. PhD thesis, College of Computing, Georgia Institute of Technology, 1992. Technical report GIT-CC-92/43.
25. E. Rissland, D. Skalak, and M.T. Friedman. Heuristic harvesting of information for case-based argument. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pages 36–43, Seattle, WA, July 1994. AAAI.
26. Uriel Rosenthal, Michael Charles, and Paul Hart, editors. *Coping with crises: The management of disasters, riots, and terrorism*. C.C. Thomas, Springfield, IL, 1989.
27. R.C. Schank. *Dynamic Memory: A Theory of Learning in Computers and People*. Cambridge University Press, Cambridge, England, 1982.
28. A. Segre. On the operationality/generality tradeoff in explanation-based learning. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, Milan, Italy, August 1987. IJCAI.
29. K. Sycara. Using case-based reasoning for plan adaptation and repair. In J. Kolodner, editor, *Proceedings of the Case-Based Reasoning Workshop*, pages 425–434, Palo Alto, 1988. DARPA, Morgan Kaufmann, Inc.
30. M. Veloso. *Planning and Learning by Analogical Reasoning*. Springer Verlag, Berlin, 1994.

31. R. Wilensky. Knowledge representation—a critique and a proposal. In J. Kolodner and C. Riesbeck, editors, *Experience, Memory and Reasoning*, chapter 2, pages 15–28. Lawrence Erlbaum Associates, 1986.