Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society, 1995

Combining Rules and Cases to Learn Case Adaptation

David B. Leake
Computer Science Department
Indiana University
Bloomington, IN 47405
leake@cs.indiana.edu

Abstract

Computer models of case-based reasoning (CBR) generally guide case adaptation using a fixed set of adaptation rules. A difficult practical problem is how to identify the knowledge required to guide adaptation for particular tasks. Likewise, an open issue for CBR as a cognitive model is how case adaptation knowledge is learned. We describe a new approach to acquiring case adaptation knowledge. In this approach, adaptation problems are initially solved by reasoning from scratch, using abstract rules about structural transformations and general memory search heuristics. Traces of the processing used for successful rule-based adaptation are stored as cases to enable future adaptation to be done by case-based reasoning. When similar adaptation problems are encountered in the future, these adaptation cases provide task- and domain-specific guidance for the case adaptation process. We present the tenets of the approach concerning the relationship between memory search and case adaptation, the memory search process, and the storage and reuse of cases representing adaptation episodes. These points are discussed in the context of ongoing research on DIAL, a computer model that learns case adaptation knowledge for case-based disaster response planning.

Introduction

The fundamental principle of case-based reasoning (CBR) for problem-solving is that new problems are addressed by *retrieving* stored records of prior problem-solving episodes and *adapting* their solutions to fit new situations. In most case-based reasoning systems, the case adaptation process is guided by fixed case adaptation rules. Practical experience developing CBR systems has shown that it is difficult to establish appropriate case adaptation rules (e.g., Allemang, 1993; Leake, 1994). In defining adaptation rules, a key problem is the classic operationality/generality tradeoff that was first observed in research on explanation-based learning (e.g., Segre, 1987): Specific rules are easy to apply and are reliable, but only apply to a narrow range of adaptation problems; abstract rules span a broad range of potential adaptations but are often hard and expensive to apply because they do not provide task- and domain-specific guidance. In those CBR systems that do perform case adaptation, specific rules are often used, requiring that the developer perform difficult analysis of the task and domain to determine which rules will be needed. In practice, the problems of defining adaptation rules are so acute that many CBR applications simply omit case adaptation (e.g., Barletta, 1994).

This paper presents a new method by which a case-based reasoning system can learn adaptation knowledge from experience. The method models a progression from case adaptation using general adaptation rules to case adaptation by a case-based reasoning process that reflects both (1) the specific adaptation problems the reasoner has encountered and (2) the reasoner's idiosyncratic memory organization and domain knowledge. The approach is motivated both by pragmatic considerations and cognitive modeling concerns. The pragmatic benefits of learning from experience to refine case adaptation knowledge are simplified knowledge acquisition, improved efficiency of adaptation, and improved quality of the results of adaptation. The benefits from a cognitive modeling perspective are in providing an account of how human abilities for adapting cases might develop and improve.

In our approach, two types of general rules are provided as the initial basis for case adaptation: rules describing structural transformations (i.e., characterizing ways to transform case structure, such as adding or substituting particular components of a case), and rules about how to search memory for the information needed to apply a transformation (e.g., that can be used to provide general guidance about how to search for appropriate items to substitute after a substitution transformation has been selected). As adaptation problems are solved successfully using these rules, two types of cases are stored to enable future case-based reasoning about the adaptation process itself. *Memory search cases* encapsulate information about the steps in the memory search process. *Adaptation cases* encapsulate information about the adaptation problem as a whole and how it was solved, including both the transformation used and the memory search process followed. Storage and reuse of these two types of cases facilitates future case adaptation in two ways. First, when a new adaptation problem is similar to a previously solved problem, the previous adaptation case is retrieved and used to suggest a transformation and memory search plan that were effective in the past. Second, even when a new adaptation problem is different from previously-solved problems, prior memory search cases can help to guide the memory search needed for the new problem. This approach to case adaptation models how case-based reasoning systems can make the transition from adaptation by general rules (which may be unreliable and hard to apply) to adaptation that reflects specific lessons acquired with adaptation experience. Experience also facilitates the solution of novel adaptation problems, because the rule-based adaptation process can apply prior memory search cases when solving new adaptation problems.

We begin with a brief discussion of the evidence for developmental changes in human case adaptation ability and an

outline of the background for our approach. We then identify and discuss key issues in the context of an implementation of this approach to learn to improve case adaptation during case-based reasoning for disaster response planning.

Perspective

Human development of case adaptation: Multiple psychological studies have shown evidence for human case-based reasoning both in the early phases of learning a domain and after achieving expertise (see Kolodner (1993) for an overview of these results). However, the development process for knowledge used to guide the application of prior cases has received less study. Gentner (1988) has shown that as children develop, a shift occurs in how they adapt stories to fit new characters. In Gentner's experiments, children first acted out stories, using toys to play the roles of the characters, and then were asked to act out the same stories but using different toys representing new characters. Although both older children (8-10 years old) and younger children (5–7 years old) were influenced by the transparency of the object mappings between toys when they choose which characters to substitute into particular roles of the stories, considerations of structural features helped the older children to make better substitutions. Our research models how criteria for deciding which adaptations to favor may be learned from experience.

Computer models of adaptation learning: Some CBR systems have capabilities for learning special-purpose case adaptation rules. For example, CHEF (Hammond, 1989), a case-based planner, augments a static library of domain-independent plan repair strategies by learning *ingredient critics* for suggesting adaptations appropriate to particular ingredients. The resulting learning is useful, but in a limited context. Another approach is to rely on an external source to provide a library of adaptation cases to be reused (Berger, 1995; Sycara, 1988). This approach is also useful, but does not address how to generate the cases used. Our aim is a general model of how an adaptation system can acquire adaptation cases for reuse as it solves novel adaptation problems.

Combining Rules and Cases to Learn Adaptation

Our approach begins with adaptation based on general rules. As novel adaptation problems are solved, the adaptation component stores information about successful adaptation episodes in a library of specific *adaptation cases* for future use.[1] Unlike abstract adaptation rules, adaptation cases encapsulate the system's experience on specific adaptation and memory search problems and reflect the system's specific task, domain, and memory organization. When no relevant cases are available, rule-based adaptation is used. Thus our method models how a reasoner can shift between rule-based and case-based case adaptation as appropriate to respond to familiar or novel adaptation problems.

[1] Useful information could also be obtained by failure-driven learning from failed adaptation attempts. That process is a future research direction.

The computer model

A project to investigate this adaptation learning method is now being conducted at Indiana University with Andrew Kinley and David Wilson. We are investigating the process in the context of adaptation of *response plans* for natural and man-made disasters. Human disaster managers are trained by studying casebooks of disasters and responses, and it has been observed that they frequently report that their new solutions are based on response plans for similar previous episodes (e.g., Rosenthal et al., 1989).

Our computer model, DIAL (for **D**isaster response with **I**ntrospective **A**daptation **L**earning), starts with a library of disaster response cases. As it generates disaster response plans, it learns both new cases and strategies for adapting its cases to fit new situations.

The entire DIAL system includes a schema-based story understander that processes inputs in a conceptual representation, a response plan retriever/instantiator, a simple evaluator for candidate response plans, and an adaptation component to adapt retrieved plans when problems are found. All components except for the adaptation component are based in a straightforward way on previous systems (e.g., the understander is based on previous understanding systems such as SAM (Cullingford, 1978), and the case-based planner is based on previous case-based planners such as CHEF (Hammond, 1989)). Consequently, further discussion will focus only on the case adaptation process.

DIAL's adaptation process

DIAL's adaptation component receives as input an instantiated disaster response plan and a description of a problem in that plan to repair by case adaptation. Its processing combines reasoning from scratch with case-based reasoning, in the following sequence of processing steps:

1. **Case-based adaptation:** DIAL attempts to retrieve cases representing adaptations that have applied to similar adaptation problems in the past. If it is successful, the adaptation case is re-applied. Otherwise, rule-based adaptation is initiated.
2. **Rule-based adaptation:** Based on the problem type, the system selects a transformation (e.g., substitution), and generates a *knowledge goal* (Ram, 1987) for the information needed for the transformation. Introspective planning (Hunter, 1990) is done to search for the needed information. Search terminates when the information is found or when a pre-set limit on the allowed number of plan steps is exceeded.
3. **Evaluation:** The adaptation is evaluated by a human user who inputs to the system whether the adaptation is acceptable. If not, other adaptations are tried. If no plans succeed with a given resource limit, rule-based adaptation is continued with an increased limit. (This gives preference to shorter memory search plans.)
4. **Storage:** When the results of adaptation are successful, the resulting response plan, the adaptation case, and the memory search plan are stored to be available for future use.

The following sections describe the rule-based adaptation and case-based adaptation processes. This is followed by a description of processing for an implemented program example.

Rule-based adaptation

In order to reason about adaptation problems, a uniform framework is needed for characterizing case adaptation. DIAL's rule-based case adaptation process is based on a characterization of the case adaptation process as involving two parts: *structural transformations* and *memory search* to find the information needed to apply the transformations. In accordance with this view, case adaptation knowledge can be treated as having two parts, abstract transformations and memory search strategies. This characterization of adaptation knowledge follows the principle of the *adaptation strategies* developed by Kass (1994). The aim of adaptation strategies is to achieve both operationality and generality by extending traditional adaptation rules to contain domain-independent strategies for searching memory to find the domain-specific information required by particular adaptation problems.

Kass's adaptation strategies were static; they were hand-coded rather than learned. However, his basic framework suggests a view of how to learn adaptation knowledge. In this view, learning specifics about case adaptation knowledge can be seen as learning the memory search information needed to operationalize general structural transformations (additions, deletions, and substitutions). Viewing adaptation learning in this way provides a broadly-applicable framework for characterizing case adaptations: it is well known that a small set of transformations is sufficient to characterize a wide range of adaptations (Carbonell, 1983; Kolodner, 1993). The resulting learning can have a significant effect on adaptation performance, because in general, a large amount of domain-specific reasoning may be required to find the information to apply those transformations. The following sections discuss how DIAL's rule-based adaptation process finds the information needed to apply general transformations when solving novel adaptation problems. This process involves selecting a transformation to apply, generating knowledge goals for the information needed to apply the transformation, and using a planning process to guide search through memory for the needed information.

Selecting transformations and generating related knowledge goals: When DIAL uses rule-based adaptation to process a novel adaptation problem, it first selects a transformation to apply. It then performs memory search for the information needed to apply the transformation. The current implementation uses a very simple scheme for selecting transformations: candidate transformations for repairing particular types of adaptation problems are indexed directly under elements in the vocabulary that the system uses for describing types of adaptation problems. Given a problem description as input, DIAL's rule-based adaptation process retrieves the transformations associated with the problem description and attempts to apply them. When reasoning from scratch, DIAL has no guidance about which to favor; the transformations associated with the problem category are applied in an arbitrary order until a successful adaptation is found. However, when an entire successful adaptation case is stored, the combination of a particular transformation and particular memory search strategy that were previously successful will be retrieved and reapplied. Consequently, the learning process involves learning about combinations of transformations and memory search strategies that are applicable to particular types of problems.

The following example illustrates this process of transformation selection. Consider a case-based disaster planner that attempts to re-apply a response plan for a chemical spill, but finds that one part of the retrieved plan is inapplicable: the retrieved plan used city buses to evacuate victims, but the city faced by the current spill has no bus system. The problem of an unavailable filler suggests using a *substitution* transformation to replace bus transportation by a different alternative.

Once a transformation has been selected, memory search is needed—in the transportation example, it is necessary to find a form of transportation to substitute. This is addressed by first representing the needed information as a knowledge goal, and then reasoning introspectively about possible plans for searching memory to find the needed information.

Representing knowledge goals: Knowledge goals have previously been investigated largely in context of opportunistically recognizing needed information as it becomes available. Representations of knowledge goals developed for this task describe the desired information in a single *concept specification* (Ram, 1987)—a pattern to be matched against new information, and a description of how the information will be used. That type of description is sufficient for its intended purpose of representing questions to be compared to new input information. However, work on DIAL suggests that when knowledge goals are used to guide active memory search, descriptions of knowledge goals must include two additional facets as well.

The first of these is what we call a *comparative specification.* The comparative specification describes how to choose between multiple items in memory that match the concept specification. The comparative specification is needed because, in a rich memory, a number of candidate items may satisfy the requirements for retrieval. For example, in searching for a substitute method of transportation for an evacuation, a comparative specification might provide the additional information that the method found should be the one yielding the highest evacuation rate.

The second type of additional information needed in knowledge goals for active memory search is what we call *search prioritization* information. Rather than only representing the search target as a complete pattern to match, DIAL's knowledge goals include a component representing information about where it is believed that relevant information could be found, i.e., about how to seek the information. There are generally many ways of searching for a single concept, involving focusing on different parts of the concept at different points in the search. As a simple example, suppose that the search goal is to find the union that someone (say John) belongs to. One strategy is to search for unions in memory, and then, for each union, to retrieve information about its members to see if John belongs to it. A better strategy might be to search for information about John's job, and then to search for likely unions given his employment. Part of DIAL's memory search planning involves determining how to break up a concept specification into a sequence of specific concepts to examine in memory. Successful stored memory search cases reflect information about which ways of prioritizing goals have proven effective. Finally, information is needed about the level of

resources to commit to the memory search process. Thus DIAL's knowledge goals include a concept specification, a comparative specification, search prioritization information, information about the amount of effort allowed in the search, and how to use the information that is found.

The planning process: DIAL's memory search process is modeled on the query reformulation process first used in CYRUS (Kolodner, 1984). It differs from CYRUS, however, in applying the process within the framework of *knowledge planning* (Hunter, 1990). In this knowledge planning framework, similar to the introspective reasoning and learning framework proposed by Kennedy (1995), a planner reasons introspectively about explicitly represented knowledge goals and how to satisfy them using internal "mental" operators. For example, two operators that can be used to guide memory search are "To find a cause for an actor's state, search for an action performed by the actor that could cause that state"; "To find actions performed by an actor, check the actor's habitual actions." Such rules are similar in flavor to the types of rules investigated in research on query transformation for information retrieval, and our aim of learning to refine memory search shares the goal of recent work in information retrieval that studies how to learn which retrieval strategies are most useful (e.g., Baudin, Pell, & Kedar, 1994).

DIAL's adaptation component is provided with a set of general memory search heuristics and basic local knowledge about its memory organization (e.g., as information on how to retrieve abstractions, on the relationships between schemas and their subparts, etc.). We note that some of these rules are relatively unguided "weak methods" of memory search (e.g., ascending and descending abstraction hierarchies to find related nodes), whose results are then filtered by constraints from the knowledge goals being satisfied. However, as is described in the following section of case-based adaptation, the aim is to generate much more effective knowledge: successful results of this relatively unguided process form the basis of specific adaptation cases that are stored to provide more precise guidance in similar future situations.

In order to be able to reason about memory search, a system must be able to reason about the meanings of its own memory links. Rather than simply being names, DIAL's memory links are structures associated with information about the relationships that hold between the linked objects, making it possible to reason about the meanings of the links in terms of those relationships. This allows the system to decide which links to follow to satisfy knowledge goals that were not anticipated when a memory was originally organized.

The planning process used by DIAL is inspired by Firby's (1989) RAPS model of reactive planning. The choice of a reactive model to guide memory search may seem surprising; reactive planning models are often advocated as a way to plan in dynamic and imperfectly-understood worlds, while the "mental" world is modeled as entirely under the reasoner's control and available for examination. However, a central difficulty with guiding the memory search process is that the general rules used to suggest memory search paths are not guaranteed to be correct in any particular instances, and in a rich memory, the costs of exhaustive examination of the information are prohibitive. Consequently, there are strong reasons for using a planning model that defers commitment to particular details of a plan and that is robust when problems arise.

Learning by storing adaptation cases

Why use CBR for adaptation learning?

Once a novel adaptation problem has been solved successfully, the question arises of what should be learned from the results of problem-solving. Our first effort at modeling the adaptation learning as operationalizing abstract transformations was the program AL (Adaptation Learner) (Leake, 1994a). AL's process for performing adaptations from scratch was similar to the process described above. After AL formed an appropriate memory search plan, it performed explanation-based generalization (EBG) (e.g., Mitchell et al., 1986) to form a new general memory search rule for future use.

One of the lessons learned from AL was that EBG is in fact inappropriate for learning the memory search task. EBG depends on a correct domain theory—to apply EBG to the memory search task, it requires a domain theory accounting for each piece of information in memory and how each piece of information is organized. Unfortunately, the memory search problem is precisely the problem of how to search memory effectively *without* such a theory. Memory search must apply heuristics to find needed information in an idiosyncratic memory whose contents and organization may not be characterized precisely. Consequently, chains of memory search rules that work in one instance may not apply to other problems that appear to be within the scope of those rules. Because case-based reasoning about adaptation retains the specifics of particular problems, it enables the lessons from memory search during prior adaptation episodes to be reused more effectively.

Representing case adaptation episodes

After completing rule-based case adaptation, DIAL stores two types of information about the adaptation episode, in two types of cases. *Adaptation cases* encapsulate an entire adaptation episode: the response plan that was retrieved, the problem that required adaptation, and the entire memory search plan used to retrieve the needed information. These cases suggest effective combinations of transformations and memory search strategies for particular adaptation problems. The second class of cases, *memory search cases,* stores information about the memory search process alone. These cases form building blocks for more effective memory search when novel adaptation problems are being solved. An open question concerns the tradeoffs in making sub-parts of the memory search cases accessible as separate cases, along the lines of Redmond's (1992) *snippets.*

Re-applying stored adaptation knowledge

The flexibility of CBR systems to address new problems comes from their ability to adapt prior cases to apply to new situations. However, a conflicting concern is the potential cost of adapting the current case compared to retrieving and applying a different case, or simply generating a new solution from scratch. DIAL controls this cost in two ways. First, the amount of memory search effort allowed when adapting adaptation cases is limited (this limit is implemented as a limit on the number of search rules applied). Second, the adaptation that is done for the memory search

cases themselves is very limited. DIAL's adaptation of memory search cases is restricted to operations such as *extracting* a sub-plan that addresses only the knowledge goal of interest from a memory search plan that addressed more knowledge goals, *adding filtering steps* to check the results of plans that address the desired knowledge goals but omit needed constraints, and *adding local search* for other nearby memory nodes when the result of memory search fails to satisfy some of the needed constraints. An important question to address is how additional search knowledge affects overall processing cost (Minton, 1988).

An extended program example

In the current implementation, DIAL's initial case library contains two disaster response plans: a response plan for an air quality disaster and a response plan for an industrial chemical spill. Starting from this case library, the system has been tested on four stories exercising different parts of its adaptation mechanisms. To illustrate its processing, we consider one of these in more detail. The stories and episodes are based on case studies from *INvironment,* a newsletter for indoor air quality consultants.

The example we consider 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.* DIAL's understander identifies the situation as involving an air quality problem, a type of disaster, and its retriever attempts to retrieve and apply a response plan for a similar disaster. The response plan it retrieves as most similar addresses the following episode: *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.*

Many of the steps in the retrieved response plan for the A & D factory disaster—investigating the extent of the risk, moving the victims to a new location, etc.—apply to the school problem as well. However, in the factory response, one of the steps is to notify the employees' union. Simple instantiation suggests notifying the union of the new victims—the schoolchildren—as well. The evaluator detects that schoolchildren do not have unions, and initiates case adaptation to repair the problem. (The problem is detected by a pattern-based anomaly detection process that compares new role-fillers in the response plan to standard expectations. This evaluation and problem characterization process is similar to that described in Leake, 1992).

DIAL's adaptation component receives two inputs describing this situation: the response plan for the A & D Manufacturing problem, instantiated to apply to the new situation, and a description of the problem for adaptation to repair: that the step notifying the students' union is not reasonable, because students do not belong to unions.

Because DIAL starts with no adaptation cases, adaptation of the response plan must be done starting from scratch, using the rule-based process. The first step necessary is to select a transformation to apply. The problem of schoolchildren being inappropriate members of a union is an instance of the problem category "role/filler mismatch." (For a description of possible problem types, see Leake, 1992.) The two transformations associated with "role/filler mismatch" are to either substitute a new filler (e.g., consider notifying the union of some other group), or to substitute a different concept in which the children play a role with relevant similarities (e.g., notifying another group relevant to the children). Many adaptations based on these two transformations are possible, but a common suggestion from readers of the story is that a reasonable substitution is to notify the children's parents. The point of DIAL's adaptation process is both to generate this answer (as one of many candidates) and to learn from its success that this is a reasonable adaptation to apply to similar future problems.

Substituting a new role corresponds to a knowledge goal to answer the question "who should be notified instead?" To specify the knowledge goal, DIAL first determines the constraints on reasonable substitutions. To find the constraints, it must hypothesize the factors that were important in the relationship between workers and their union in the A & D manufacturing problem.

DIAL infers constraints to consider by formulating different possible "views" (Wilensky, 1986) of the relationship between the union and the workers in the original episode. Each of these views suggests aspects of union membership that might have been relevant and consequently are potential candidates for aspects to preserve when making the substitution. One candidate relationship involves representation: *being represented by the union.* This suggests generating the knowledge goal of finding representatives of the children. Memory search for representatives of children locates "parents" as a possibility. This is not the only candidate—other groups such as "student government" are also considered, but are rejected in the evaluation phase. In the future, the resulting successful adaptation (replacing the union by parents) will be retrieved and reapplied.

The effects of learning and the ability to reuse adaptation cases are demonstrated by DIAL's processing of the following stories. One, involving a chemical spill at a school, prompts retrieval of the stored case from a prior chemical spill story and its response plan. However, the previous plan cannot be applied in its entirety because it (like the A & D plan) involves notifying the students' union. However, the stored adaptation case from the Beaver Meadow school example can be reapplied in a straightforward way. This example illustrates the benefits of learning not only domain cases (here, disaster response cases) but also adaptation cases: Learning adaptation cases increases the effectiveness of applying existing cases to new problems.

Another implemented example shows how DIAL can reuse a stored adaptation case in a more flexible way. In that example, the story being processed involves an air quality problem on a military base. The most similar prior case is the A & D air quality disaster. However, when DIAL attempts to apply the response plan from that case, there is a problem similar to the problem that arose when generating the response plan for the Beaver Meadow school disaster: soldiers do not have unions. In this situation, the learned adaptation case (substituting parents for unions) does not apply directly. Nevertheless, the problem is similar, and DIAL uses the prior case as

a starting point for future reasoning, reusing the initial part of its search chain. Because *representation* was relevant in determining whom to substitute in the adaptation for the students, the previous case suggests searching for representatives in the current situation. DIAL searches its memory for representatives of soldiers and finds "commanding officers" as a possible group to notify. This demonstrates the ability to perform some adaptation of adaptation cases. More study is needed to identify appropriate adaptation strategies and their utility in terms of both the cost of adaptation and the quality of the results produced.

Conclusions

Providing appropriate case adaptation knowledge is a difficult problem in developing case-based reasoning systems. Our research addresses that problem with a method for learning specific adaptation knowledge. DIAL builds a library of adaptation cases from specific episodes of applying general adaptation rules. The adaptation cases provide more effective guidance to adaptation by allowing reuse of successful adaptation strategies for the system's particular domain, task, and memory organization.

Our current research presents a new answer to the question of how specific case adaptation knowledge is acquired. It has developed methods for generating knowledge goals for case adaptation, a representation for the needed knowledge goals, and answers to questions about the nature of adaptation knowledge and how it may be re-applied. The next phases of the research include extending and testing the model on additional examples, investigating the role of failure-driven learning, and evaluating the effects of case learning on the quality and efficiency of the case adaptation process, especially as the number of adaptation and memory search cases grows.

Acknowledgments

This work was supported by the National Science Foundation under Grant No. IRI-9409348. Andrew Kinley and David Wilson made very helpful comments on a draft of this paper.

References

Allemang, D. (1993). Review of the first European workshop on case based reasoning. *Case-Based Reasoning Newsletter*, *2*(3). Electronic newsletter of AK-CBR.

Barletta, R. (1994). A hybrid indexing and retrieval strategy for advisory CBR systems built with ReMind. In *Proceedings of the Second European Workshop on Case-Based Reasoning*, pp. 49–58 Chantilly, France.

Baudin, C., Pell, B., & Kedar, S. (1994). Using induction to refine information retrieval strategies. In *Proceedings of the twelfth national conference on artificial intelligence*, pp. 553–559 Seattle, WA.

Berger, J. (1995). Using past repair episodes. Manuscript.

Carbonell, J. (1983). Learning by analogy: formulating and generalizing plans from past experience. In Michalski, R., Carbonell, J., & Mitchell, T. (Eds.), *Machine Learning: An Artificial Intelligence Approach*. Tioga Publishing Company, Cambridge, MA.

Cullingford, R. (1978). *Script Application: Computer Understanding of Newspaper Stories*. Ph.D. thesis, Yale University. Computer Science TR # 116.

Firby, R. (1989). *Adaptive Execution in Complex Dynamic Worlds*. Ph.D. thesis, Yale University. Computer Science Department TR 672.

Gentner, D. (1988). Metaphor as structure mapping: the relational shift. *Child Development*, *59*, 47–59.

Hammond, K. (1989). *Case-Based Planning: Viewing Planning as a Memory Task*. Academic Press, San Diego.

Hunter, L. (1990). Planning to learn. In *Proceedings of the Twelfth Annual Conference of the Cognitive Science Society*, pp. 261–268 Cambridge, MA.

Kass, A. (1994). Tweaker: adapting old explanations to new situations. In Schank, R., Riesbeck, C., & Kass, A. (Eds.), *Inside Case-Based Explanation*, chap. 8, pp. 263–295. Lawrence Erlbaum Associates.

Kennedy, A. (1995). 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.

Kolodner, J. (1984). *Retrieval and Organizational Strategies in Conceptual Memory*. Lawrence Erlbaum Associates, Hillsdale, NJ.

Kolodner, J. (1993). *Case-Based Reasoning*. Morgan Kaufmann, San Mateo, CA.

Leake, D. (1992). *Evaluating Explanations: A Content Theory*. Lawrence Erlbaum Associates, Hillsdale, NJ.

Leake, D. (1994a). Towards a computer model of memory search strategy learning. In *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*, pp. 549–554 Atlanta, GA.

Leake, D. (1994b). Workshop report: the AAAI-93 workshop on case-based reasoning. *AI Magazine*, *15*(1), 63–64.

Minton, S. (1988). *Learning Search Control Knowledge: An Explanation-Based Approach*. Kluwer, Boston.

Mitchell, T., Keller, R., & Kedar-Cabelli, S. (1986). Explanation-based generalization: a unifying view. *Machine Learning*, *1*(1), 47–80.

Ram, A. (1987). AQUA: asking questions and understanding answers. In *Proceedings of the Sixth Annual National Conference on Artificial Intelligence*, pp. 312–316 Seattle, WA. Morgan Kaufmann Publishers, Inc.

Redmond, M. (1992). *Learning by Observing and Understanding Expert Problem Solving*. Ph.D. thesis, Georgia Inst. of Technology. TR # GIT-CC-92/43.

Rosenthal, U., Charles, M., & Hart, P. (Eds.). (1989). *Coping with crises: The management of disasters, riots, and terrorism*. C.C. Thomas, Springfield, IL.

Segre, A. (1987). On the operationality/generality tradeoff in explanation-based learning. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence* Milan, Italy.

Sycara, K. (1988). Using case-based reasoning for plan adaptation and repair. In Kolodner, J. (Ed.), *Proceedings of the Case-Based Reasoning Workshop*, pp. 425–434 Palo Alto. DARPA, Morgan Kaufmann, Inc.

Wilensky, R. (1986). Knowledge representation—a critique and a proposal. In Kolodner, J. & Riesbeck, C. (Eds.), *Experience, Memory and Reasoning*, chap. 2, pp. 15–28. Lawrence Erlbaum Associates.