The Omnipresence of Case-Based Reasoning in Science and Application

David W. Aha
Navy Center for Applied Research in Artificial Intelligence
Naval Research Laboratory, Code 5510
Washington, DC 20375 USA
aha@aic.nrl.navy.mil
http://www.aic.nrl.navy.mil/~aha

Abstract

A surprisingly large number of research disciplines have contributed towards the development of knowledge on *lazy problem solving*, which is characterized by its storage of ground cases and its demand driven response to queries. Case-based reasoning (CBR) is an alternative, increasingly popular approach for designing expert systems that implements this approach. This paper lists pointers to some contributions in some related disciplines that offer insights for CBR research. We then outline a small number of Navy applications based on this approach that demonstrate its breadth of applicability. Finally, we list a few successful and failed attempts to apply CBR, and list some predictions on the future roles of CBR in applications.

1 Case-Based Reasoning

Case-based reasoning (CBR) is a multi-disciplinary subject that focuses on the reuse of experiences (i.e., *cases*). It is difficult to find consensus on more detailed definitions of CBR because it means different things to different groups of people. For example, consider its interpretation by the following three groups:

- *Cognitive Scientists*: CBR is a plausible high-level model for cognitive processing (Kolodner, 1993a).
- *Artificial Intelligence Researchers*: CBR is a computational paradigm for problem solving (Aamodt & Plaza, 1994).
- *Expert System Practitioners*: CBR is a design model for expert systems that can be used in either stand alone or embedded architectures (Watson, 1997).

To introduce CBR requires identifying it in a particular context. For example, variants of Aamodt and Plaza's (1994) problem-solving cycle are frequently used when introducing CBR to AI researchers. Figure 1 displays a view of five top-level steps that, combined, input a problem description and output a (proposed) solution.

1. *Retrieve*: Given a problem, retrieve a set of stored cases (e.g., $\langle$problem,solution,outcome$\rangle$ triplets) whose problems are judged as similar.
2. *Reuse*: Apply one or more solutions from these retrieved cases, perhaps by combining them with each other or with other knowledge sources.
3. *Revise*: Adapt the retrieved solution(s), as needed, in an attempt to solve the new problem.
4. *Review*: Evaluate the outcome(s) when applying the constructed solution to the current problem. If the outcome is not acceptable, then the solution will require further revision.
5. *Retain*: Consider adding the new triplet of problem, revised solution, and outcome to the library as a new case.

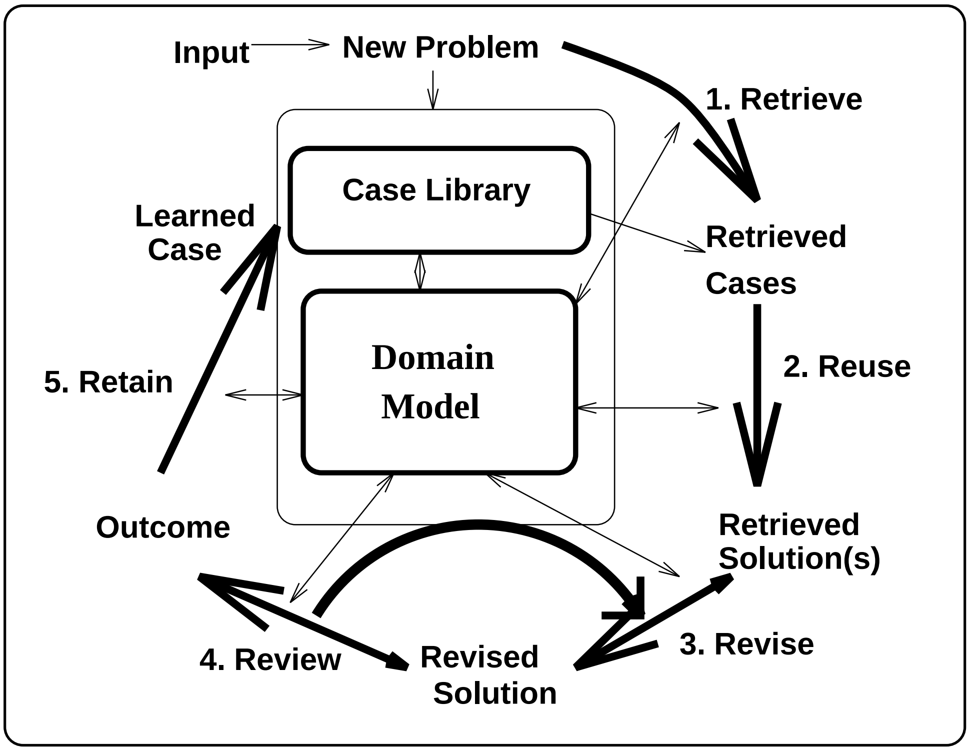

Figure 1: A View of the Case-Based Reasoning Problem Solving Cycle

From the AI perspective, many of today's most popular CBR research directions interact with this cycle, involving either methods for refining these top-level steps or in embedding CBR in a multimodal reasoning architecture.

The cognitive view of CBR was first popularized by Roger Schank and his colleagues during the late 1970's and 1980's at Yale University (Schank & Abelson, 1977; Riesbeck & Schank, 1989). This lead to three USA DARPA workshops, in 1988, 1989, and 1991, and several AAAI-sponsored workshops and symposia during 1988-1993. The German CBR community held their first annual CBR workshop in 1992, the Europeans in 1993, and the United Kingdom in 1995; these meetings signaled an explosive growth in interest in CBR, but one that focused on artificial intelligence and expert systems rather than cognitive science. Recent AAAI-sponsored CBR meetings (1994-1998) have also focused research more on computer science than on cognitive science, as have the first two international meetings (Veloso & Aamodt, 1995; Leake & Plaza, 1997).

Similarly, while some recent books contain material devoted to cognitive models (e.g., Schank & Kass, 1994; Leake, 1996), most recent books related to CBR have stressed issues in computer science, including design (Maher et al., 1995; Gebhardt et al., 1997; Maher & Pu, 1997), expert systems (Lewis, 1995; Watson, 1997), machine learning (Kolodner, 1993b; Aha, 1997), and applications (Watson, 1997).

Several other research disciplines (e.g., machine learning, cognitive psychology, process planning, statistics) have also contributed to a growing body of knowledge on reasoning from cases. However, few CBR publications relate these other contributions to CBR research. This brief paper, based on an invited talk at the 1997 SGES International Conference on Knowledge Based Systems and Applied Artificial Intelligence, begins in Section 2 by surveying some research related to CBR in other fields, and how that research can benefit the development of CBR approaches. We also describe a small set of Navy projects that exemplify a variety of roles in which CBR can be applied for expert systems tasks. Next, we discuss some successes and failures in practicing CBR. Finally, we list some focus areas in which CBR approaches are expected to contribute.

2 The Omnipresence of Case-Based Reasoning

The basic notions of case-based reasoning (e.g., storage of experiences, similarity computations, indexing) are familiar to researchers in several disciplines. Thus, CBR researchers are in a unique position to learn from advances in these disciplines, which may prove an advantage to those studying alternative strategies for designing expert systems (e.g., rule-based reasoning). Furthermore, there are a large variety of roles in which CBR techniques can assist in expert systems. However, we are not arguing that the complete CBR problem-solving cycle, especially that of symbolic solution adaptation, is frequently studied outside of CBR. Rather, we argue that a distinctive characteristic of CBR, that of *lazy problem solving*, is what ties CBR research to several other disciplines.

Lazy problem solving is a form of problem solving where computation is performed on a demand-driven basis. We define purely lazy problem solvers to display the following three behavior: (all beginning with "D")

1. *Defer*: They do not process their inputs (i.e., data) until given information requests.
2. *Data-driven*: They respond to requests by combining information from the stored data.
3. *Discard*: They dismiss any temporary intermediate results created during problem solving.

In contrast, *eager* algorithms compile their inputs into an intensional data structure (i.e., discarding their inputs), reply to information requests using this a priori compiled abstraction, and retain it for future requests. For example, in the context of supervised learning algorithms, the k-nearest neighbor classifier is a lazy algorithm, while algorithms that greedily induce decision trees (e.g., C4.5 (Quinlan, 1993a)) are eager.

This lazy/eager distinction exhibits many interesting tradeoffs. For example, the following benefits of lazy problem solving approaches exist:

1. *Elicitation*: Lazy approaches require the availability of cases rather than difficult-to-extract rules. (This is also true for most machine learning approaches.) This can significantly refocus knowledge acquisition efforts on how to structure cases.
2. *Problem Solving Bias*: Because cases are in raw form, they can be used for several different problem solving purposes. In contrast, rules and other abstractions can generally be used for only the purpose that guided their compilation.
3. *Incremental Learning*: Lazy approaches typically have low training (i.e., data processing) costs in comparison with approaches that attempt to compile data into concise abstractions. However, the tradeoff often exists that lazy approaches require more work to answer information queries, although smart caching schemes can be used to decrease this workload (e.g., Clark & Holte, 1992).
4. *Disjunctive Solution Spaces*: Lazy approaches are often most appropriate for tasks whose solution spaces are complex, making them less appropriate for approaches that replace data with abstractions (Aha, 1992).
5. *Precedent Explanations*: By virtue of storing rather than discarding case data, lazy approaches can generate precedent explanations (i.e., based on the retrieved cases). Characteristic (i.e., abstract) explanations, if requested, can always be derived from the stored set of cases in a demand-driven manner.

Table 1: Some Research Areas Related to Lazy Problem Solving

Research Area	Topic
Cognitive Psychology	Exemplar models
Pattern Recognition	Edited k-nearest neighbor classifiers
Machine Learning	More than just instance-based approaches
Cognitive Science	CBR, Analogy
Information Retrieval	Document retrieval
Statistics/Robotics	Locally weighted regression
Data Structures	Recent variants on k-d trees
Software Engineering	Software reuse
Process Planning	Variant process planning

6. *Sequential Problem Solving*: Sequential tasks often benefit from the storage of a history in the form of the states that lead to the current state. Lazy approaches are used to store this information, which can then be used, for example, to disambiguate states (e.g., McCallum, 1995).

However, perhaps the most compelling reason for using a lazy problem solving approach is that it is highly intuitive; experts often relate their problem solving behavior in ways that suggest a form of case-based reasoning. We discuss these and other benefits of lazy approaches throughout this section. See (Aha, 1997) for a collection of articles focussed on lazy learning.

Lazy problem solving is a distinguishing characteristic of CBR approaches: they perform demand-driven reasoning from a stored library of cases. We will use this perspective when describing the impact of research in other disciplines on CBR research in Section 2.1, and also when exemplifying some roles of CBR techniques in applications in Section 2.2.

2.1 Omnipresence in Science

There are a variety of ways in which research contributions on reasoning from cases, from several related disciplines, can benefit performance tasks, and to promote awareness of this body of knowledge. For example, some research in the areas listed in Table 1 clearly relate to CBR. Due to time constraints, we only briefly describe some of this research here with the objective of providing pointers to relevant research.

2.1.1 Cognitive Psychology

Related research in cognitive psychology has focused on human concept representation. Some CBR researchers have claimed that algorithms based on nearest neighbor retrieval have support in human subject studies. This claim is misleading. While Smith and Medin's (1981) book, which has inspired several CBR researchers, argued that support exists for "exemplar models," they were referring to algorithms based on the probability choice model, in which similarities from all stored cases contribute towards categorization predictions based on their similarity to the new "probe." Medin and Schaffer's (1978) Context Model was based on this approach. Implicit in their model was what Shepard (1987) called the *Universal Law of Generalization*, which states that the probability that human subjects will confuse two stimuli is an exponentially decreasing function of their distance (i.e., in a *psychological space*). Shepard showed how this law accounted for 30 years of human subject data, and it was Nosofsky (1986) who, in his Generalized Context Model, clarified how the Context Model employed this assumption. Medin, Nosofsky, their colleagues, and several others have reported enormous support for the psychological plausibility of models that claim humans base categorization (and other) decisions on stored cases (i.e., "exemplars"). For example, Hintzman and Ludlam (1984) showed that this approach, embodied in Minvera II, accounts for prototypicality effects, among others. And Krushke (1992) showed how exemplar models can account for yet more behaviors in human subject data in a neural network architecture based on radial basis functions. This is one of many demonstrations, across many disciplines, of how CBR behavior can be captured in alternative problem-solving architectures.

This area is rich in topics covered, including modeling selective attention via feature weighting, context-dependent classification, correlated features, and frequent empirical comparisons with models that use other representations for concepts. However, for several decades, it was inundated by studies that focused on cognitively plausible representation models for concepts (e.g., prototype, feature frequency, exemplar). Researchers would show that an extension of their favorite model would outperform (i.e., correlate better with subject data) than previous models using alternative representations (i.e., by introducing ever more numbers of free parameters in their models, and assuming that a process model existed that would set these parameters to values that could allow the model to closely correlate with subject behavior). This would be followed by promising extensions of the other models. In fact, these competitions were futile; as Barsalou (1989) noted, exemplar models can be designed to closely imitate abstraction models, and visa versa. The emphasis of study should be on the combination of process model and representation rather than on representation alone. This should alert CBR researchers to the fact that they can design their approaches to take advantage of advances in eager problem-solving methodologies, and that lazy variants of these approaches also exist.

2.1.2 Pattern Recognition

Related pattern recognition research on lazy algorithms has, as in cognitive psychology, frequently focused on classification. However, pattern recognition research on this topic began much earlier, and it focused on process models in addition to representation issues. For example, Fix and Hodges (1951) are generally credited with discovering the simple k-nearest neighbor (k-NN) case-based classifier, which predicts that a new case's class is determined as a function of its k most similar stored cases.

CBR researchers can benefit from studies in pattern recognition on *case deletion*. Several researchers have investigated variants of edited k-NN classifiers. For example, one research line followed Wilson's (1972) lead, in which incorrectly classified cases were deleted (e.g., Wagner, 1973; Koplowitz & Brown 1981). This research emphasized the introduction of algorithms whose probability of error more closely approached the Bayes optimal rate than previous algorithms. The intuition was that these classifiers deleted only atypical cases, and that they could deliver reasonable accuracy for both typical and atypical cases. Alternatively, Hart (1968) popularized the study of algorithms that deleted correctly classified cases, arguing that the atypical cases provided more information on class boundaries. This line of research has been more empirically inclined, with a focus on thresholding (Sebestyen, 1962), iterative deletion (Gates, 1973), and domain-specific feature selection (Kurtzberg, 1987). Naturally, a combination of these two strategies has also been investigated in detail (e.g., Voisin & Devijer, 1987). Dasarathy's (1991) collection contains several classic papers related to this research.

Interest in the AI community on case deletion is strong. For example, the 1995 IJCAI best paper award was given to a paper on this topic (Smyth & Keane, 1995). Thus, it is probable that several of the advances in pattern recognition on this topic are of interest to CBR researchers. Furthermore, work by Dasarathy (1980) and others on learning when to not make predictions from stored cases should also be of interest. Advances in unsupervised approaches and data structures have also been made in this research area. Finally, pattern recognition research on learning in the limit analyses of edited k-NN classifiers should be of interest to formalists studying case-based retrieval. In general, many related and earlier contributions in pattern recognition continue to be re-invented in other communities.

2.1.3 Machine Learning

Research in machine learning (ML) on lazy problem solving has, as with both cognitive psychology and pattern recognition, frequently focused on classification. However, many of the advances on lazy learning can be modified for use in the CBR problem solving cycle, and other research advances on lazy learning exist (e.g, in problem solving, clustering). In addition, many concerns of the CBR researcher are also addressed in related ML research, such as forms of indexing, tolerating noise and missing values, feature selection, incremental updating, and related issues.

Many contributions in ML related to lazy problem solving can be roughly dichotomized according to whether they concern incorporating eager characteristics into lazy algorithms or visa versa. Additionally, several researchers have investigated less closely integrated combinations of lazy and eager approaches.

1. Eager realizations of lazy approaches:

Few ML researchers are working with purely lazy algorithms; they are needlessly memory and cpu intensive. Instead, ML research has focused on different methods for caching information with lazy algorithms.

Perhaps the earliest research on lazy learning reported at a ML meeting was by Bradshaw (1986; 1987), who introduced the phrase *instance-based* learning (IBL)[1] while working on his 1985 dissertation. He focused on the task of recognizing spoken letters, and introduced an instance-averaging algorithm named *disjunctive spanning* that attempted to locate centroids of disjuncts. Kibler and Aha (1987) later introduced *IB2*, a similar algorithm that performs instance filtering (i.e., by retaining only correctly classified cases) rather than instance averaging (Kibler & Aha, 1988). Both of these algorithms were, not surprisingly, highly similar to earlier research on pattern recognition. In particular, disjunctive spanning is closely realted to Sebestyen's (1962) algorithm (i.e., but without pre-determined thresholds), and IB2 is a simplification of CNN (Hart, 1968).[2]

Since these early efforts, lazy algorithms have undergone dramatic design enhancements. Researchers have studied several lazy learning topics, including

- case selection (Zhang et al., 1997),
- concept distribution skew (Cardie & Howe, 1997),
- concept shift (Salganicoff, 1997),
- cost-sensitive learning (Tan & Schlimmer, 1990; Turney, 1993),
- discretization (Ting, 1997; Wilson & Martinez, 1997a),

[1] Several synonyms have been used to describe these algorithms. For example, these include {case,exemplar,instance,memory}-based {learning,reasoning}. These names reflect a focus on representation to the exclusion of processing. This is one reason why we prefer the phrase *lazy learning*; it implies both.

[2] CNN was also rediscovered elsewhere. For example, see (Kurtzberg, 1987).

- feature selection and weighting (Kelly & Davis, 1991; Cain et al., 1991; Cardie, 1993; Skalak, 1994; Ricci & Avesani, 1995; Kohavi et al., 1997; Domingos, 1997; Ling & Wang, 1997; Maron & Moore, 1997; Wettschereck et al., 1997; Howe & Cardie, 1997),
- information theory (Lee, 1994; Cleary & Trigg, 1995; Wettschereck & Dietterich, 1995),
- noise (Stanfill, 1987; Aha & Kibler, 1989; Aha et al., 1991; Ting, 1997),
- parallel implementations (Stanfill & Waltz, 1986),
- preference learning (Branting & Broos, 1994),
- speedup techniques (Deng & Moore, 1995; Grolimund & Ganascia, 1996; Daelemans et al., 1997),
- storage reduction (Zhang, 1992; Cameron-Jones, 1992; Skalak, 1994; Wilson & Martinez, 1997b),
- symbolic features (Cost & Salzberg, 1993; Biberman, 1994), and
- voting techniques (Alpaydin, 1997; Skalak, 1997; Ricci & Aha, 1997).

This is an incomplete list, and we do not attempt to relate these approaches here. Most publications focused on improvements concerning speed, accuracy, storage, and/or scope of applicability.

Some of the most interesting research on this topic has investigated the use of alternative representations with lazy approaches. For example, Salzberg (1991), Wettschereck & Dietterich (1995), and Domingos (1995), among others, have investigated algorithms that generalize cases to rules. Elliot and Scott (1991) examined how to define similarity using hierarchical structures. Emde and Wettschereck (1996) described a lazy algorithm for first-order case representations, while Langley et al. (1997) examine a matrix representation of probabilities for cases.

A significant amount of ML research on lazy algorithms has focused on robotic control tasks (e.g., Connell & Utgoff, 1987; Maes & Brooks, 1990; Moore, 1990). See (Atkeson et al., 1997a) for an excellent survey on this subject, and its relationship to statistics. They strongly advocate a fairly pure lazy approach where an impressive amount of parameter tuning is performed during prediction tasks. An important lesson for CBR researchers is that the set of parameters typically considered in AI research on lazy learning is often a small subset of the possible parameters considered in research on statistics.

Several researchers have described formal analyses of lazy algorithms, most of which have been inspired by the PAC learning framework (e.g., Albert & Aha, 1991; Lin & Vitter, 1994). Turney (1994) analyzed cross-validation error and voting, while several researchers have now addressed average case studies (e.g., Langley & Iba, 1993; Griffiths, 1996; Okamoto & Yugami, 1997; see Griffith's dissertation for additional references, in particular his work with colleagues). Fewer formal studies have yet to appear on case retrieval, adaptation, and other issues related to CBR research. However, those who endeavor to pursue such analyses would do well to consider the contributions of this line of research.

2. Lazy realizations of eager approaches:

An interesting branch of research, only recently pursued, is that of investigating the benefits of using a lazy approach for what have previously been thought of as eager algorithms. For example, Friedman (1994), Smyth and Cunningham (1994), and Friedman et al. (1996) have described the benefits of using a lazy approach for decision tree induction. The primary advantage is that the trees (really, paths) created by these algorithms are query-specific (Atkeson et al., 1997a); the only features in the tree are those pertinent to the query and missing values are never considered. Other benefits include increased classification accuracy. Webb (1996), among others, are now investigating a similar idea for rule-induction algorithms. The lesson for CBR researchers is that query-specific approaches for CBR may impact the design of indexing strategies.

Planning can also be viewed as a lazy task. For example, Epstein and Shih (1997) demonstrated that lazy reasoning using sequences of cases can improve planning performance for a bridge playing task.

3. Loose integrations of lazy and eager approaches:

This section describes a few examples of less-closely coupled integrations between eager and lazy learning approaches. We categorize these contributions according to the eager data structure employed in these integrations.

Decision trees: Several ML researchers have explored the benefits of integrating lazy and eager approaches to enhance learning performance. For example, Indurkhya and Weiss (1995) and Torgo (1997), among others, have investigated decision tree induction where lazy algorithms are used at the leaves. Ting (1994) examined the utility of lazy learning for small disjuncts, where a decision tree was used to determine where small disjuncts existed and to answer queries otherwise. Quinlan (1993b) examined how decision tree predictions could be used to modify the prediction of a lazy learner. Utgoff (1989), among others, has examined how decision trees can be efficiently incrementally updated by storing cases at its leaves. This is reminiscent of how unsupervised algorithms, such as COBWEB (Fisher, 1989), use evaluation functions to update their hierarchies. Similar approaches might prove useful for guiding the incremental modifications of indexing structures for case retrieval.

Rule induction: Widmer's (1993) description of how a rule-induction can be used to cluster sets of cases is related to similar research on clustering using decision trees, as mentioned above. Zhang and Michalski (1995) instead described an extension of the AQ approach for rule induction in which specific cases were stored with the rule set.

Genetic algorithms: Some researchers have described feature weighting and selection approaches for lazy algorithms using genetic algorithms (e.g., Kelly & Davis, 1991; Skalak, 1994; Wilson & Martinez, 1996). Sheppard and Salzberg (1997) instead investigated how a genetic algorithm can provide good examples for a lazy classifier in the context of a missile evasion task.

Neural networks: Volper and Hampson (1987) described early research on the utility of using specific instance nodes in multilayer neural networks. Lazy classifiers have a close relationship with radial basis networks (Moody & Darken, 1988; Broomhead & Lowe, 1988), which employ Gaussian, rather than sigmoidal, hidden units that are fixed on a specific location in the instance space.

Bayesian reasoning: Breese and Heckerman (1995) describe how a three-level case-based Bayesian network can be used for diagnosis tasks, but do not specifically focus on learning issues. Similarly, Tirri et al. (1996), among others, have described a Bayesian framework for CBR. However, fewer integrations of these approaches than might be expected have focused on learning issues.

Reinforcement learning: Moore and Atkeson (1993) described the use of prioritized sweeping to speed up the reinforcement learning process. The general idea is to focus learning effort as guided by previous interesting experiences. In contrast, McCallum (1995) explained how storing case histories of exploration traces can help to solve state aliasing problems.

Analytic learning: Some research on explanation-based learning has examined the utility of lazy approaches for deduction (e.g., Tadepalli (1989); Clark & Holte (1992)). More recently, Borrajo and Veloso (1997) have shown that a lazy learning approach can be used to efficiently acquire control knowledge in an incremental planning process.

2.1.4 Summary

This section briefly described investigations of research related to CBR in a few disciplines, but neglected research in many others. For example, closely related research has been pursued in cognitive science on computational analogy (Gentner & Forbus, 1991), in information retrieval on inference networks (Callan et al., 1995), in physics on provably correct retrieval, similarity, and adaptation functions for a set of domains (Rudolph & Hertkorn, 1997), in software engineering on software reuse, and in statistics on recursive partitioning (Friedman, 1994) and tangent distance functions (e.g., Hastie and Tibshirani, 1994). An enormous amount of research on reasoning from cases has appeared in research published in many disciplines.

A few application areas are particularly good matches for the biases of lazy approaches. For example, research on natural language learning (e.g., Daelemans et al., 1997) often involves working with data sets whose concepts are highly disjunctive. In such circumstances, it is well known that lazy approaches have benefits vs. eager approaches (e.g., Aha, 1992). This suggests further study of lazy realizations of eager approaches for natural language learning tasks. Another good application area concerns robotic control. See (Atkeson et al., 1997b) for a survey on selected lazy approaches for control tasks.

Given this discussion of related research, several lessons suggest how to decide whether to use case-base vs. alternative reasoning approaches. For example:

- *Incremental learning*: Case-based approaches are suited for incremental learning tasks because they prevent the premature selection of summary abstractions. For more information, see (Matheus, 1987).
- *Highly disjunctive spaces*: Case-based approaches are particularly suited for tasks where abstractions tend to yield over-generalizations. For example, in highly disjunctive spaces (e.g., several natural language processing tasks), eager approaches often combine disjuncts representing different solutions, and thereby will reduce task performance.
- *Sequence-based reasoning*: Temporal reasoning from sequences has proven effective for such diverse tasks as selecting moves in two-person games (Epstein & Shih, 1997) and for disambiguating states (McCallum, 1995). The potential of CBR for other types of sequential tasks is worth investigating.
- *Query-specific reasoning*: Lazy approaches provide the enormous benefit of using local information to characterize states and generate predictions (e.g., Friedman et al., 1996). Because abstraction approaches attempt to fit all cases simultaneously, they may lose information that is crucial for specific queries.
- *Training speed*: Case-based approaches are usually much cheaper to use when storing information. Thus, CBR is preferable for tasks where storage speed is valued more highly than query-response speed, and/or when efficient caching mechanisms can be used to reduce the time to respond to subsequent information requests.

- *Missing values*: Lazy approaches, whether for use in a case-based, decision tree, or other algorithm, are useful for tolerating missing values because lazy methods exist that require processing only the values known for the given query.
- *Precedent explanations*: By retaining specific cases, decision aids systems that reason from these cases can always use them to explain system behavior. This is not possible with approaches that discard cases, yet characteristic explanations (i.e., based on abstractions from the cases) can always be generated from a set of cases upon demand.

However, CBR approaches have their limitations. For example, they tend to have larger storage requirements, and demand attention for the definitions of case retrieval and adaptation. We strongly encourage that practitioners use a comparative analysis of the pros and cons for whether a CBR approach is well-suited for their task.

2.2 Omnipresence in Application

Case-base approaches have been applied to a broad spectrum of problems in various roles. Readers interested in details should consult (Watson, 1997) or certain CBR-related WWW pages (e.g., AI-CBR, U. Kaiserslautern's page, or our own). Rather than repeating what others have written, we instead discuss a small number of Navy CBR applications, focusing on the different roles that case-based models can play.

2.2.1 Feature selection

Bankert and Aha (1996) describe an eager realization of a lazy approach for feature selection. Bankert (1994) had previously reported an application of a traditional feature selection algorithm for a cloud classification task. He used a clustering algorithm to evaluate the quality of a feature subset, along with a neural network classifier. Although he reported what seemed to be reasonable performance, we found that a *wrapper approach* (John et al., 1994), in which the same algorithm was used for both the feature subset evaluator and the classifier, was preferable. In particular, we used a case-based classifier for both components, and it significantly increased classification accuracies. Thus, the algorithm *cached* information as to what features to focus on during classification. Thus, it is not purely lazy, but rather is an eager realization of a lazy approach. A lesson we learned is that the low training costs required by a CBR approach are particularly useful in a feature selection algorithm, which must evaluate a large number of feature subsets and, therefore, training speed is a concern. This eliminated any hope of using an expensive eager approach.

More recently, we examined the use of an alternative output representation for these tasks (Aha & Bankert, 1997). In particular, we found that *error-correcting output codes* (ECOCs), a type of distributed representation, can further enhance accuracy, though at the cost of speed and storage. It is feasible that ECOCs can also be of use to case retrieval tasks (i.e., to assist in distinguishing cases during retrieval), although this has not yet been investigated.

2.2.2 Robotic Navigation

Grefenstette (1996) describes a loose integration of lazy and eager problem solving approaches for a robotic navigation task. He focused on anytime learning issues, where the world's state was available through sensor inputs. Previous research had shown that genetic algorithms could learn rules to control the robots behavior (i.e., its actuators), and they could be used to guide the robot to solve navigation tasks. However, Grefenstette was interested in developing an extension of this research where sensor failures could be tolerated. He pursued this by learning one rule population per world state, where a state corresponded to a subset of the sensors that were working properly. A monitor was available to indicate changes in sensor availability, and was used to trigger the algorithm to select the top-performing rules from appropriate populations.

Thus, Grefenstette viewed the multiple rule populations as a case library, where each case was a population. His approach selected and imported parts of each case to the "current" rule-learning population as triggered by the world state monitor. This case-based approach allowed the robot to gracefully recover from sensor failures, or other changes in the world state, significantly more quickly than when multiple rule populations were not cached. This approach is being pursued in ongoing projects at the Naval Research Laboratory's Center for Applied Research in Artificial Intelligence.

2.2.3 Interactive Troubleshooting

Naval personnel working on weapons systems often have challenging troubleshooting tasks. In particular, some systems have several volumes dedicated to troubleshooting scenarios. These are difficult to master, particularly when short billets dictate that personnel have little time to master the machinery they work with.

Conversational CBR approaches (Aha & Breslow, 1997) have been used to assist naval personnel with troubleshooting complex systems. One application works as follows: Onboard personnel contact one of the four Fleet Technical Service Centers with troubleshooting questions. Personnel at the center have access to a conversational CBR tool that, given answers concerning the situation, can walk them through to a solution. This requires only a short conversation where the system troubleshooter answers a few system-prompted questions. At any point in the conversation, the system permits the troubleshooter to answer any of a small set of questions, thus allowing the system to continue processing even when some pertinent answers are unknown.

A fielded version of this approach has been in use since 1995. It required the construction of a conversational case library, which included some cached information, such as relevance weights on questions. Between two and ten calls per week are handled by Naval personnel at the Centers. This is a successfully applied example of an eagerly realized lazy approach.

2.2.4 Summary

In addition to the three Navy CBR applications mentioned above, some ongoing projects are also targeting the use of a CBR approach for Navy applications. For example, one involves comparing two records of processing in an empirical software testing task. The task is to determine whether the revised module exhibits any significant behavioral anomalies. Another application will involve using conversational CBR technology to train users on how to operate a transmission electron microscope.

3 Successes and Failures of Case-Based Reasoning

This section briefly describes some successful and failed CBR ventures. We concentrate on real-world experiences rather than those that are purely academic, in large part because these experiences usually contain lessons that suggest pursuing important research directions.

3.1 Successes

This section lists three categories of tasks where CBR has been successfully applied. For each category, we mention a company whose products focus on these tasks. For more information on other companies who are pursuing these and other interesting tasks (e.g., AcknoSoft), and on other successful CBR ventures, please see (Watson, 1997).

3.1.1 Interactive Troubleshooting

By far the most extensive application of CBR technology has been in the area of interactive troubleshooting. In particular, Inference Corporation (www.inference.com) has over 650 corporate contracts with organizations who are using Inference's *CBR Content Navigator* products for help desk and WWW self-help applications. These products are based on the conversational CBR (CCBR) approach that was mentioned in Section 2.2.3, and have been extremely successfully applied. For example, on the first day of Broderbund's release of *Riven*, their followup of the extremely popular PC game *Myst*, approximately 3000 clients used Broderbund's WWW-mounted self-help server to answer their troubleshooting questions rather than phone Broderbund for help. Thus, this application helps Broderbund to lower its cost for call center support.

One reason for Inference's success is that they have targeted a niche that is a good fit for CBR. In remote troubleshooting tasks, clients often can answer some information concerning their problem, but perhaps not all information. Thus, an approach that prompts them with only one question (e.g., a standard decision tree approach) is often not suitable for these clients. In contrast, this is a design focus of CCBR tools.

Perhaps the most significant concern regarding CCBR systems is that they demand a learning curve for authoring case libraries. Our group's research focuses on simplifying the case authoring task for CCBR systems. We created a system, named *NaCoDAE* (Navy Conversational Decision Aids Environment), for this purpose (Breslow & Aha, 1997). We are pursuing specific methods for simplifying case authoring:

1. *Library Revision*: Aha and Breslow (1997) describe *Clire*, a component of NaCoDAE that enforces case authoring guidelines for a given case library. It does this by revising the case library, editing the cases by first re-representing them in a modified decision tree hierarchy, which simplifies the editing process. We found that the Clire-transformed libraries we tested showed significant increases in retrieval precision and efficiency. This decreases the burden on case authors to design case libraries, thus simplifying their task.

2. *Question Weighting*: Few methods for weighting questions exist for case retrieval algorithms other than those used for classification tasks. We have investigated a limited linear programming approach to automatically assign weights, and are comparing it to a more promising method based on extracting weights from a Clire tree. By eliminating the need to assign question weights, we simplify the task of the case author.

3. *Dialogue Inferencing*: CCBR systems begin by inputting a natural language text description of a user's problem. Unfortunately, current methods for interpreting that text do not easily relate it to the case library. In particular, the text typically implicitly answers some questions in the case library. Although some CCBR systems permit users to insert rules that relate text to questions, and relate questions to other questions, these rules can be huge in number. We advocate a different strategy whereby the case author interactively guides the system in building a model of the objects in the case library and a model of how the questions relate to the object model. Given the current problem description (i.e., the user's text and their

answered questions), these models can be used by a relational database retrieval engine to answer other questions in the case library. We are currently examining how Parka can support this process (Aha & Maney, 1997).

More information on our CBR research activities can be viewed at

www.aic.nrl.navy.mil/~aha/cbr/practical-advances.html.

3.1.2 Recommenders

Firefly (www.firefly.com) is one of the first companies to begin marketing software development tools that use a CBR component to make recommendations to clients. For example, consider an application that is supposed to recommend books for you to read. The system would begin by inserting some of your likes and dislikes, based on your past reading experiences and interests. It then would consult a large case library of similar inputs from other book readers, along with detailed facts concerning available books. The system could then respond to you with a set of suggested books, derived from similar reading preferences exhibited in the case library. Not surprisingly, Barnes and Noble developed an application of Firefly's *Passport Office* system that has this functionality.

Recommender systems are not limited to suggesting book preferences. For example, they could be used for more challenging tasks, such as electronic dating services. Kris Hammond's group is exploring similar ideas at the University of Chicago (see www.cs.uchicago/groups/ai).

3.1.3 Internet Commerce

TecInno (www.tecinno.de) is focusing on developing products that are closely related to recommender systems, but are geared for internet commerce tasks. In particular, their *CBR-Works Online* tool has been used for such tasks as sales support, consulting systems, and product catalogues. CBR-Works extracts product requirements from clients, uses them to retrieve product announcements, and revises them according to a model of the client's interests. This model is updated according to feedback the model provides on the product announcements retrieved for them. Thus, learning user models is a distinguishing characteristic of this knowledge-intensive approach to case-based reasoning.

3.2 Failures

This short list of failures concerning applying CBR should be familiar to expert systems researchers and practitioners.

3.2.1 Corporate Support

The most important objective of any AI application should be to obtain corporate support for the project. Without it, otherwise successful applications will not flourish. For example, Marc Goodman (1994) created an application of projective visualization for Nestle/UK's coffee roaster machines. These expensive machines were suffering from downtime costs when sensor readings indicated that they had to be shut down to prevent irreparable damage. At such times, the machines had to be diagnosed and their current contents (i.e., slop) had to be discarded.

To reduce this problem, Goodman's system compared the current state of a coffee machine to previously recorded states, and also had access to a simulator so that it could predict future states. It would notify users whenever sensor problems were anticipated with high probability. This application was well-received by its users; it was responsible for greatly reducing the costs of using these roasters.

Unfortunately, corporate support was not behind this project due to corporate politics. The result was that it killed the near-term possibility of future CBR applications at Nestle/UK. Kitano et al. (1992) emphasize the necessity of corporate support in large-scale CBR projects.

3.2.2 Knowledge Acquisition

CBR is not a magic bullet for the expert systems community. It is a technology that demands attention to the process of *case engineering*, which bears resemblance to *knowledge engineering*. However, the emphasis here is on developing case libraries that can deliver solutions to clients. While some commercial CBR tools can sometimes be successfully applied to client tasks "out of the box," more often then not the clients recognize that case authoring skill is required to create a successful CBR application.

Under those circumstances, clients have, in general, three options. First, they can turn to the CBR company for consulting services. This is an expensive proposition. Second, they can develop in-house expertise, as has been done at many client companies. However, some companies do not have need for such expertise, and decide to pursue the third option: abandon the CBR application. Not surprisingly, several client companies have chosen this third option. Thus, simplifying the case authoring task is of great practical value to prospective clients of commercial CBR tools.

3.2.3 Scope of Applicability

Finally, a third CBR failure concerns mismatching the capabilities of a CBR tool with its intended task. Some tasks are such that they do not match well with CBR tools. For example, they may not draw on the benefits of CBR (see Section 2), and other tools might be more practical for the task. This has made some potential client companies wary of investing in CBR technology. For those clients, their best strategy is to become informed about the types of problems for which CBR is particularly suited, and to examine whether their problems have similar characteristics. Given that CBR approaches have a particularly broad scope of application, we anticipate that analyses of this kind will often reveal that a CBR component has a useful role in many industry and government problem solving applications.

4 Predictions for Case-Based Reasoning

The CBR research and commercial communities are thriving. Given their current activities, we can predict some trends for future CBR applications.

1. *Continuing Current Trends*: Applications in help-desk, self-service, and recommender systems (i.e., including internet commerce) will thrive. The continuing focus will be on reducing user requirements (e.g., by simplifying the case authoring tasks), broadening problem-solving abilities by integrating case-based with eager approaches, and on using learning to model (and subsequently exploit) user preferences.

2. *Information Retrieval*: Cases represent structured information, at least with respect to comparatively unstructured data sources such as the WWW, document collections, and many databases. Thus, there will be increased activity on extracting information in less structured forms and recording them as cases. This caching approach will simplify answering future,

similar information requests. Inference Corporation, among others, is pursuing this strategy in their recent product designs.

3. *System Monitoring*: There is an increased need to solve sequential problem solving tasks by recording history information. CBR is well-suited to assist in these tasks; comparisons can be made between histories to direct decision aids activities.
4. *Knowledge Management*: A primary concern of corporate entities is in recording and reusing the knowledge of their employees. The motivation is clear; employee turnover means loss of knowledge, and thus loss of intellectual capital (Stewart, 1997). This is an excellent applications area for case-based reasoning technology that is only now beginning to be pursued.

Naturally, given the content of this talk, a remaining prediction must be made: there will be an increase in interdisciplinary research (and perhaps applications) between case-based reasoning and other disciplines. For example, various advances in CBR could influence the directions of research on variant process planning, information retrieval, and machine learning. Likewise, contributions could be drawn from rather than exported to these communities.

5 Summary

This paper briefly presented an argument that case-based reasoning (CBR) research is not alone in the universe. Rather, the lazy problem solving model that underlies the CBR problem-solving cycle is endemic to many research disciplines, although only parts of the cycle are typically of interest in those disciplines. Furthermore, there are a large assortment of ways in which case-based approaches can contribute to an integrated multimodal reasoning architecture. We briefly discussed some of the benefits of using a CBR approach, CBR successes, failures, and future trends. For more information, we encourage readers to examine the references and, perhaps more importantly, review the compiled WWW sites on CBR (e.g., AI-CBR, the University of Kaiserslautern's pages, or our own).

Acknowledgements

This research was supported by a grant from the Office of Naval Research. This hastily-written report was based on the invited academic keynote address at the 1997 Specialist Group on Expert Systems International Conference on Knowledge-Based Systems and Applied Artificial Intelligence.

Bibliography

Aamodt, A., & Plaza, E. (1994). Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications*, *7*, 39–59.

Aha, D. W. (1992). Generalizing from case studies: A case study. In *Proceedings of the Ninth International Conference on Machine Learning* (pp. 1–10). Aberdeen, Scotland: Morgan Kaufmann.

Aha, D. W. (Ed.) *Lazy Learning*. Norwell, MA: Kluwer Academic Publishers.

Aha, D. W., & Bankert, R. L. (1997). Cloud classification using error-correcting output codes. *Artificial Intelligence Applications: Natural Science, Agriculture, and Environmental Science*, *11*(1), 13–28.

Aha, D. W., & Breslow, L. A. (1997). Refining conversational case libraries. *Proceedings of the Second International Conference on Case-Based Reasoning* (pp. 267–278). Providence, RI: Springer.

Aha, D. W., & Kibler, D. (1989). Noise-tolerant instance-based learning algorithms. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence* (pp. 794–799). Detroit, MI: Morgan Kaufmann.

Aha, D. W., Kibler, D., & Albert, M. K. (1991). Instance-based learning algorithms. *Machine Learning*, *6*, 37–66.

Aha, D. W., & Maney, T. (1997). *A model-based approach for supporting dialogue inferencing in a conversational case-based reasoner* (Technical Report AIC-97-023). Washington, DC: Naval Research Laboratory, Navy Center for Applied Research in Artificial Intelligence.

Albert, M. K., & Aha, D. W. (1991). Analyses of instance-based learning algorithms. In *Proceedings of the Ninth National Conference on Artificial Intelligence* (pp. 553–558). Anaheim, CA: AAAI Press.

Alpaydin, E. (1997). Voting over multiple condensed nearest neighbors. *Artificial Intelligence Review*, *11*, 115–132.

Atkeson, C., Moore, A., & Schaal, S. (1997a). Locally weighted learning. *Artificial Intelligence Review*, *11*(1–5), 11–73.

Atkeson, C., Moore, A., & Schaal, S. (1997b). Locally weighted learning for control. *Artificial Intelligence Review*, *11*(1–5), 75–113.

Bankert, R. L. (1994). Cloud classification of AVHRR imagery in maritime regions using a probabilistic neural network. *Journal of Applied Meteorology*, *33*, 909–918.

Bankert, R. L., & Aha, D. W. (1996). Improvement to a neural network cloud classifier. *Applied Meteorology*, *35*, 2036–2039.

Barsalou, L. W. (1989). On the indistinguishability of exemplar memory and abstraction in category representation. In T. K. Srull & R. S. Wyer (Eds.), *Advances in social cognition.* Hillsdale, NJ: Lawrence Erlbaum.

Biberman, Y. (1994). A context similarity measure. In *Proceedings of the European Conference on Machine Learning* (pp. 49–63). Catania, Italy: Springer-Verlag.

Borrajo, D., & Veloso, M. (1997). Lazy incremental learning of control knowledge for efficiently obtaining quality plans. *Artificial Intelligence Review*, *11*, 371–405.

Bradshaw, G. L. (1986). Learning by disjunctive spanning. In T. M. Mitchell, J. G. Carbonell, & R. S. Michalski (Eds.), *Machine learning: A guide to current research.* Boston, MA: Kluwer.

Bradshaw, G. (1987). Learning about speech sounds: The NEXUS project. In *Proceedings of the Fourth International Workshop on Machine Learning* (pp. 1–11). Irvine, CA: Morgan Kaufmann.

Branting, L. K., & Broos, P. (1994). Automated acquisition of user preferences. *International Journal of Human-Computer Studies, 46*, 55–77.

Breese, J. S., & Heckerman, D. (1995). Decision-theoretic case-based reasoning. *Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics* (pp. 56–63). Ft. Lauderdale, FL: Unpublished.

Breslow, L., & Aha, D. W. (1997). *NaCoDAE: Navy Conversational Decision Aids Environment* (Technical Report AIC-97-018). Washington, DC: Naval Research Laboratory, Navy Center for Applied Research in Artificial Intelligence.

Broomhead, D. S., & Lowe, D. (1988). Multivariable functional interpolation and adaptive networks. *Complex Systems*, *2*, 321–355.

Cain, T., Pazzani, M. J., & Silverstein, G. (1991). Using domain knowledge to influence similarity judgement. In *Proceedings of the Case-Based Reasoning Workshop* (pp. 191–202). Washington, DC: Morgan Kaufmann.

Callan, J. P., Lu, Z., & Croft, W. B. (1995). Searching distributed collections with inference networks. *Proceedings of the Eighteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 21–28). Seattle, WA: Unpublished.

Cameron-Jones, R. M. (1992). Minimum description length instance-based learning. In *Proceedings of the Fifth Australian Joint Conference on Artificial Intelligence* (pp. 368–373). Hobart, Australia: World Scientific.

Cardie, C. (1993). Using decision trees to improve case-based learning. In *Proceedings of the Tenth International Conference on Machine Learning* (pp. 25–32). Amherst, MA: Morgan Kaufmann.

Cardie, C., & Howe, N. (1997). Improving minority-class prediction using case-specific feature weights. *Proceedings of the Fourteenth International Conference on Machine Learning* (pp. 57–65). Nashville, TN: Morgan Kaufmann.

Clark, P., & Holte, R. (1992). Lazy partial evaluation: An integration of explanation-based generalisation and partial evaluation. In *Proceedings of the Ninth International Conference on Machine Learning* (pp. 82–91). Aberdeen, Scotland: Morgan Kaufmann.

Cleary, J. G., & Trigg, L. E. (1995). K*: An instance-based learner using an entropic distance measure. In *Proceedings of the Twelfth International Machine Learning Conference* (pp.108–114). Tahoe City, CA: Morgan Kaufmann.

Connell, M. E., & Utgoff, P. E. (1987). Learning to control a dynamic physical system. In *Proceedings of the Sixth National Conference on Artificial Intelligence* (pp. 456–460). Seattle, WA: Morgan Kaufmann.

Cost, S., & Salzberg, S. (1993). A weighted nearest neighbor algorithm for learning with symbolic features. *Machine Learning*, *10*, 57–78.

Daelemans, W., van den Bosch, A., & Weijters, G. (1997). IGTree: Using trees for compression and classification in lazy learning algorithms. *Artificial Intelligence Review*, *11*, 407–423.

Dasarathy, B. V. (1980). Nosing around the neighborhood: A new system structure and classification rule for recognition in partially exposed environments. *Pattern Analysis and Machine Intelligence*, *2*, 67–71.

Dasarathy, B. V. (Ed.). (1991). *Nearest neighbor(NN) norms: NN pattern classification techniques.* Los Alamitos, CA: IEEE Computer Society Press.

Deng, K., & Moore, A. W. (1995). Multiresolution instance-based learning. In *Proceedings of the International Joint Conference on Artificial Intelligence* (pp. 1233–1239). Montreal: Morgan Kaufmann.

Domingos, P. (1995). Rule induction and instance-based learning: A unified approach. *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence* (pp. 1226–1232). Montreal, Canada: Morgan Kaufmann.

Domingos, P. (1997). Context-sensitive feature selection for lazy learners. *Artificial Intelligence Review, 11*, 227–253.

Elliot, T., & Scott, P. D. (1991). Instance-based and generalization-based learning procedures applied to solving integration problems. In *Proceedings of the Eighth Conference of the Society for the Study of Artificial Intelligence* (pp. 256–265). Leeds, England: Springer-Verlag.

Emde, W. & Wettschereck, D. (1996). Relational instance-based learning. *Proceedings of the Thirteenth International Conference on Machine Learning* (pp. 122–130). Bari, Italy: Morgan Kaufmann.

Epstein, S. L., & Shih, J. (1997). Learning from sequential examples: initial results with instance-based learning. In D. Wettschereck & D. W. Aha *Working Notes for Case-Based Learning: Beyond Classification of Feature Vectors* (Technical Report AIC-97-005). Washington, DC: Naval Research Laboratory, Navy Center for Applied Research in Artificial Intelligence.

Fisher, D. H. (1989). Knowledge acquisition via incremental conceptual clustering. *Machine Learning, 2*, 139–172.

Fix, E., & Hodges, J. L., Jr. (1951). *Discriminatory analysis, nonparametric discrimination, consistency properties* (Technical Report 4). Randolph Field, TX: United States Air Force, School of Aviation Medicine.

Friedman, J. H. (1994). Flexible metric nearest neighbor classification. Unpublished manuscript available by anonymous FTP from playfair.stanford.edu (see pub/friedman/README).

Friedman, J. H., Kohavi, R., & Yun, Y. (1996). Lazy decision trees. *Proceedings of the Thirteenth National Conference on Artificial Intelligence* (pp. 717–724). Portland, OR: AAAI Press.

Gates, G. W. (1972). The reduced nearest neighbor rule. *Institute of Electrical and Electronics Engineers Transactions on Information Theory, 18*, 431–433.

Gebhardt, F., Voss, A., Graether, W., & Schmidt-Belz, B. (1997). *Reasoning with Complex Cases.* Boston: Kluwer.

Gentner, D., & Forbus, K. (1991). MAC/FAC: A model of similarity-based retrieval. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society* (pp. 504–509). Chicago, IL: Lawrence Earlbaum.

Goodman, M. (1994). Results on controlling action with projective visualization. In *Proceedings of the Twelfth National Conference on Artificial Intelligence* (pp. 1245–1250). Seattle, WA: AAAI Press.

Grefenstette, J. J. (1996). Genetic learning for adaptation in autonomous robots. In M. Jamshidi, F. Pin, & D. Dauchez (Eds.), *Robotics and Manufacturing: Recent Trends in Research and Applications.* New York: ASME Press.

Griffiths, A. D. (1996). *Inductive generalisation in case-based reasoning systems* (Technical Report YCST 97/02). Heslington, England: University of York, Department of Computer Science.

Grolimund, S., & Ganascia, J.-L. (1995). Integrating case-based reasoning and tabu search for solving optimisation problems. *Proceedings of the First International Conference on Case-Based Reasoning* (pp. 451–460). Sesimbra, Portugal: Springer-Verlag.

Hart, P. E. (1968). The condensed nearest neighbor rule. *Institute of Electrical and Electronics Engineers Transactions on Information Theory, 14*, 515–516.

Hastie, T., & Tibshirani, R. (1994). Discriminant adaptive nearest neighbor classification. Unpublished manuscript.

Hintzman, D. L., & Ludlam, G. (1984). Differential forgetting of prototypes and old instances: Simulation by an exemplar-based classification model. *Memory & Cognition*, *8*, 378–382.

Howe, N, & Cardie, C. (1997). Examining locally varying weights for nearest neighbor algorithms. *Proceedings of the Second International Conference on Case-Based Reasoning* (pp. 455–466). Providence, RI: Springer.

Indurkhya, N., & Weiss, S. M. (1995). Using case data to improve on rule-based function approximation. *Proceedings of the First International Conference on Case-Based Reasoning* (pp. 217–228). Sesimbra, Portugal: Springer-Verlag.

John, G., Kohavi, R., & Pfleger, K. (1994). Irrelevant features and the subset selection problem. In *Proceedings of the Eleventh International Machine Learning Conference* (pp. 121–129). New Brunswick, NJ: Morgan Kaufmann.

Kelly, J. D., Jr., & Davis, L. (1991). A hybrid genetic algorithm for classification. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence* (pp. 645–650). Sydney, Australia: Morgan Kaufmann.

Kibler, D., & Aha, D. W. (1987). Learning representative exemplars of concepts: An initial case study. In *Proceedings of the Fourth International Workshop on Machine Learning* (pp. 24–30). Irvine, CA: Morgan Kaufmann.

Kibler, D., & Aha, D. W. (1988). Comparing instance-averaging with instance-filtering learning algorithms. In *Proceedings of the Third European Working Session on Learning* (pp. 63–80). Glasgow, Scotland: Pitman.

Kitano, H., Shibata, A., Shimazu, H., Kajihara, J., & Sato, A. (1992). Building large-scale and corporate-wide case-based systems: Integration of the organizational and machine executable algorithms. In *Proceedings of the Tenth National Conference on Artificial Intelligence* (pp. 843–849). San Jose, CA: AAAI Press.

Kohavi, R., Langley, P., & Yun, Y. (1997). The utility of feature weighting in nearest neighbor algorithms. In M. van Someren & G. Widmer (Eds.) *Poster Papers: Ninth European Conference on Machine Learning.* Unpublished manuscript.

Kolodner, J. (1993a). *Case-based reasoning.* San Mateo, CA: Morgan Kaufmann.

Kolodner, J. (1993b). *Case-based learning.* Boston: Kluwer.

Koplowitz, J., & Brown, T. A. (1981). On the relation of performance to editing in nearest neighbor rules. *Pattern Recognition*, *13*, 251–255.

Kruschke, J. K. (1992). ALCOVE: An exemplar-based connectionist model of category learning. *Psychological Review*, *99*, 22-44.

Kurtzberg, J. M. (1987). Feature analysis for symbol recognition by elastic matching. *International Business Machines Journal of Research and Development*, *31*, 91–95.

Langley, P., & Iba, W. (1993). Average-case analysis of a nearest neighbor algorithm. *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence* (pp. 889–894). Chambery, France: Morgan Kaufmann.

Langley, P., Pfleger, K., & Sahami, M. (1997). Lazy acquisition of place knowledge. *Artificial Intelligence Review*, *11*(1–5), 315–342.

Leake, D. B. (Ed.) *Case-Based Reasoning: Experiences, Lessons, and Future Directions.* AAAI Press/MIT Press.

Leake, D. B., & Plaza, E. (Eds.) (1997). *Case Based Reasoning Research and Development: Proceedings of the First International Conference.* Providence, RI: Springer.

Lee, C. (1994). An instance-based learning method for databases: An information theoretic approach. In *Proceedings of the European Conference on Machine Learning* (pp. 387–390). Catania, Italy: Springer-Verlag.

Lewis, L. (1995). *Managing Computer Networks: A Case-Based Reasoning Approach.* Norwood, MA: Artech House.

Lin, J.-H., & Vitter, J. S. (1994). A theory for memory-based learning. *Machine Learning, 17*, 143–168.

Ling, X. C., & Wang, H. (1997). Towards optimal weights setting for the 1-nearest neighbour learning algorithm. *Artificial Intelligence Review, 11*(1–5), 255–272.

Maes, P., & Brooks, R. (1990). Learning to coordinate behaviors. In *Proceedings of the Eighth National Conference on Artificial Intelligence* (pp. 796–802). Boston, MA: Morgan Kaufmann.

Maher, M. L., Balachandran, M. B., & Zhang, D. M. (1995). *Case-Based Reasoning in Design.* Mahwah, NJ: Erlbaum.

Maher, M. L. and Pu, P. (Eds.) (1997). *Issues and Applications of Case-Based Reasoning to Design.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Maron, O., & Moore, A. W. (1997). The racing algorithm: Model selection for lazy learners. *Artificial Intelligence Review, 11*(1–5), 193–225.

Matheus, C. J. (1987). *Conceptual purpose: Implications for representation and learning in machines and humans* (Technical Report 87-1370). Urbana, IL: University of Illinois, Department of Computer Science.

McCallum, R. A. (1995). Instance-based utile distinctions for reinforcement learning. *Proceedings of the Twelfth International Conference on Machine Learning.* Lake Tahoe, CA: Morgan Kaufmann.

Medin, D. L., & Schaffer, M. M. (1978). Context theory of classification learning. *Psychological Review, 85*, 207–238.

Moody, J., & Darken, C. (1988). Learning with localized receptive fields. In *Proceedings of the 1988 Connectionist Models Summer School* (pp. 133–143). Pittsburgh, PA: Morgan Kaufmann.

Moore, A. W. (1990). Acquisition of dynamic control knowledge for a robotic manipulator. In *Proceedings of the Seventh International Conference on Machine Learning* (pp. 244–252). Austin, TX: Morgan Kaufmann.

Moore, A. W. & Atkeson, C. G. (1993). Prioritized sweeping: Reinforcement learning with less data and less time. *Machine Learning, 13*, 103–130.

Nosofsky, R. M. (1986). Attention, similarity, and the identification-categorization relationship. *Journal of Experimental Psychology: General, 15*, 39–57.

Okamoto, S., & Yugami, N. (1997). Theoretical analysis of a case retrieval method based on neighborhood of a new problem. *Proceedings of the Second International Conference on Case-Based Reasoning* (pp. 349–358). Providence, RI: Springer.

Quinlan, J. R. (1993a). *C4.5: Programs for machine learning.* San Mateo, CA: Morgan Kaufmann.

Quinlan, J. R. (1993b). Combining instance-based learning and model-based learning. In *Proceedings of the Tenth International Conference on Machine Learning* (pp. 236–243). Amherst, MA: Morgan Kaufmann.

Ricci, F., & Aha, D. W. (1997). Extending local learners with error-correcting output codes (Technical Report AIC-97-001). Washington, DC: Naval Research Laboratory, Navy Center for Applied Research in Artificial Intelligence.

Ricci, F., & Avesani, P. (1995). Learning a local similarity metric for case-based reasoning. In *Proceedings of the First International Conference on Case-Based Reasoning* (pp. 301–312). Sesimbra, Portugal: Springer-Verlag.

Riesbeck, C. K., & Schank, R. C. (1989). *Inside Case-Based Reasoning.* Hillsdale, NJ: Erlbaum.

Rudolph, S. (1997). On the foundations and applications of similarity theory to case-based reasoning. *Proceedings of the Twelvth International Conference for Applications of Artificial Intelligence in Engineering.* Capri, Italy.

Salganicoff, M. (1997). Tolerating concept and sampling shift in lazy learning using prediction error context switching. *Artificial Intelligence Review*, *11*, 133–155.

Salzberg, S. L. (1991). A nearest hyperrectangle learning method. *Machine Learning*, *6*, 251–276.

Schank, R. C., & Abelson, R. P. (1977). *Scripts, plans, goals, and understanding.* Hillsdale, NJ: Lawrence Erlbaum.

Schank, R. C., Kass, A., & Riesbeck, C. K. (Eds.) (1994). *Inside Case-Based Explanation.* Hillsdale, NJ: Erlbaum.

Sebestyen, G. S. (1962). *Decision-making processes in pattern recognition.* New York, NY: Macmillan.

Shepard, R. N. (1987). Toward a universal law of generalization for psychological science. *Science*, *237*, 1317–1323.

Sheppard, J. W., & Salzberg, S. L. (1997). A teaching strategy for memory-based control. *Artificial Intelligence Review*, *11*(1–5), 343–370.

Skalak, D. (1994). Prototype and feature selection by sampling and random mutation hill climbing algorithms. In *Proceedings of the Eleventh International Machine Learning Conference* (pp. 293–301). New Brunswick, NJ: Morgan Kaufmann.

Skalak, D. (1997). *Prototype selection for composite nearest neighbor classifiers* (Technical Report 96-89). Amherst, MA: University of Massachusetts, Department of Computer Science.

Smith, E. E., & Medin, D. L. (1981). *Categories and concepts.* Cambridge, MA: Harvard University Press.

Smyth, B., & Cunningham, P. (1994). A comparison of incremental CBR and inductive learning. In M. Keane, J. P. Haton, & M. Manago (Eds.) *Working Papers of the Second European Workshop on Case-Based Reasoning.* Chantilly, France: Unpublished.

Smyth, B. & Keane, M. T. (1995). Remembering to forget: A competence-preserving deletion policy for case-based systems. *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (pp. 377–383). Montreal, Canada: Morgan Kaufmann.

Stanfill, C. (1987). Memory-based reasoning applied to English pronunciation. In *Proceedings of the Sixth National Conference on Artificial Intelligence* (pp. 577–581). Seattle, WA: Morgan Kaufmann.

Stanfill, C., & Waltz, D. (1986). Toward memory-based reasoning. *Communications of the Association for Computing Machinery*, *29*, 1213–1228.

Stewart, T. A. (1997). *Intellectual Capital: The New Wealth of Organizations.* New York: Doubleday/Currency.

Tadepalli, P. (1989). Lazy explanation-based learning: A solution to the intractable theory problem. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence* (pp. 694–700). Detroit, MI: Morgan Kaufmann.

Tan, M., & Schlimmer, J. C. (1989). Cost-sensitive concept learning of sensor use in approach and recognition. In *Proceedings of the Sixth International Workshop on Machine Learning* (pp. 392–395). Ithaca, NY: Morgan Kaufmann.

Ting, K. M. (1994). The problem of small disjuncts: Its remedy in decision trees. In *Proceedings of the Tenth Canadian Conference on Artificial Intelligence* (pp. 91–97).

Ting, K. M. (1997). Discretization in lazy learning algorithms. In D. W. Aha (Ed.), *Lazy Learning.* Norwell, MA: Kluwer.

Tirri, H., Kontkanen, P., & Myllmäki, P. (1996). A Bayesian framework for case-based reasoning. *Proceedings of the Third European Workshop on Case-Based Reasoning* (pp. 413–427). Lausanne, Switzerland: Springer-Verlag.

Torgo, L. (1997). Kernel regression trees. In M. van Someren & G. Widmer (Eds.) *Poster Papers: Ninth European Conference on Machine Learning.* Unpublished manuscript.

Turney, P. D. (1993). Exploiting context when learning to classify. In *Proceedings of the European Conference on Machine Learning* (pp. 402–407). Vienna, Austria: Springer-Verlag.

Turney, P. D. (1994). Theoretical analyses of cross-validation error and voting in instance-based learning. *Journal of Experimental and Theoretical Artificial Intelligence*, *6*, 331–360.

Utgoff, P. E. (1989). Incremental induction of decision trees. *Machine Learning*, *4*, 161–186.

Veloso, M., & Aamodt, A. (Eds.) (1995). *Case Based Reasoning Research and Development: Proceedings of the First International Conference.* Sesimbra, Portugal: Springer.

Voisin, J., & Devijver, P. A. (1987). An application of the Multiedit-Condensing technique to the reference selection problem in a print recognition system. *Pattern Recognition*, *5*, 465–474.

Volper, D. J., & Hampson, S. E. (1987). Learning and using specific instances. *Biological Cybernetics*, *57*, 57–71.

Wagner, T. J. (1973). Convergence of the edited nearest neighbor. *Institute of Electrical and Electronic Engineers Transactions on Information Theory*, *19*, 696–697.

Watson, I. (1997). Applying Case-Based Reasoning. San Francisco: Morgan Kaufmann.

Webb, G. (1996). A heuristic covering algorithm outperforms learning all rules. *Proceedings of Information, Statistics, and Induction in Science* (pp. 20–30). Melbourne: World Scientific.

Wettschereck, D., Aha, D. W. & Mohri, T. (1997). A review and empirical comparison of feature weighting methods for a class of lazy learning algorithms. *Artificial Intelligence Review*, *11*(1–5), 273–314.

Wettschereck, D., & Dietterich, T. G. (1995). An experimental comparison of the nearest neighbor and nearest hyperrectangle algorithms. *Machine Learning*, *19*, 5–28.

Widmer, G. (1993). Combining knowledge-based and instance-based learning to exploit qualitative knowledge. *Informatica*, *17*, 371–385.

Wilson, D. (1972). Asymptotic properties of nearest neighbor rules using edited data. *Institute of Electrical and Electronic Engineers Transactions on Systems, Man and Cybernetics*, *2*, 408–421.

Wilson, D. R., & Martinez, T. R. (1996). Instance-based learning with genetically derived attribute weights. *Proceedings of the International Conference on Artificial Intelligence, Expert Systems, and Neural Networks* (pp. 11–14). Unpublished manuscript.

Wilson, D. R., & Martinez, T. R. (1997a). Improved heterogeneous distance functions. *Journal of Artificial Intelligence Research*, *6*, 1–34.

Wilson, D. R., & Martinez, T. R. (1997b). Instance pruning techniques. *Proceedings of the Fourteenth International Conference on Machine Learning* (pp. 403–411). Nashville, TN: Morgan Kaufmann.

Zhang, J. (1992). Selecting typical instances in instance-based learning. In *Proceedings of the Ninth International Machine Learning Conference* (pp. 470–479). Aberdeen, Scotland: Morgan Kaufmann.

Zhang, J., & Michalski, R. S. (1995). An integration of rule induction and exemplar-based learning for graded concepts. *Machine Learning*, *21*(3), 235–268.

Zhang, J., Yim, Y.-S., & Yang, J. (1997). Intelligent selection of instances for prediction functions in lazy learning algorithms. *Artificial Intelligence Review*, *11*(1-5), 175–192.