

Retrieval, reuse, and retention in CBR

Ramon López de Mántaras¹, David McSherry², Derek Bridge³, David Leake⁴, Barry Smyth⁵, Susan Craw⁶, Boi Faltings⁷, Mary Lou Maher⁸, Michael Cox⁹, Kenneth Forbus¹⁰, Mark Keane¹¹, Agnar Aamodt¹², Ian Watson¹³

¹*Artificial Intelligence Research Institute, CSIC, Campus UAB, 08193 Bellaterra, Spain*

e-mail: mantaras@iia.csic.es

²*School of Computing and Information Engineering, University of Ulster, Coleraine BT52 1SA, Northern Ireland, UK*

e-mail: dmg.mcscherry@ulster.ac.uk

³*Department of Computer Science, University College Cork, Ireland*

e-mail: d.bridge@cs.ucc.ie

⁴*Computer Science Department, Indiana University, Lindley Hall 215, 150 S. Woodlawn Avenue, Bloomington, IN 47405, USA.*

e-mail: leake@cs.indiana.edu

⁵*School of Computer Science and Informatics, University College Dublin, Belfield, Dublin 4, Ireland*

email: Barry.Smyth@ucd.ie

⁶*The Robert Gordon University, Aberdeen, UK*

e-mail: S.Craw@comp.rgu.ac.uk

⁷*AI-Lab, Swiss Federal Institute of Technology, CH-1015 Lausanne, Switzerland*

e-mail : Boi.Faltings@epfl.ch

⁸*School of Information Technologies, University of Sydney, Australia*

e-mail : marym@it.usyd.edu.au

⁹*BBN Technologies, Cambridge, MA 02138, USA*

e-mail: mcox@bbn.com

¹⁰*EECS Department, Northwestern University, Evanston, IL 60208, USA*

e-mail : forbus@northwestern.edu

¹¹*School of Computer Science and Informatics, University College Dublin, Belfield, Dublin 4, Ireland.*

e-mail : mark.keane@ucd.ie

¹²*Department of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway*

agnar.aamodt@idi.ntnu.no

¹³*Department of Computer Science, University of Auckland, Auckland, New Zealand*

e-mail: ian@cs.auckland.ac.nz

Abstract

Case-base reasoning (CBR) is an approach to problem solving that emphasizes the role of prior experience during future problem solving: new problems are solved by reusing the solutions to similar problems that have been solved in the past. To date CBR has enjoyed considerable

success in a wide variety of problem solving tasks and domains. In this paper, following a brief overview of the traditional problem-solving cycle in case-based reasoning, we examine the cognitive science foundations of CBR and the relationship between CBR and analogical reasoning. We then go on to review a representative selection of CBR research on aspects of case retrieval, reuse/revise, and retention in the past couple of decades.

1 Introduction

In problem solving terms, the world is often and repetitive and regular place. Similar problems tend to recur and similar problems tend to require similar solutions. Case-based reasoning techniques (CBR) attempt to exploit this repetitiveness and regularity by leveraging past problem solving experience – in the form of concrete problem solving cases – when it comes to solving new problems. In short, a case-based reasoner solves new problems by adapting solutions that were used to solve old problems (Riesbeck & Schank, 89)

Unlike the majority of AI techniques case-based reasoning is fundamentally memory-based. It appeals to our intuitions about human problem solving –that when humans solve a problem they often base their solution on one that worked for a similar problem remembered from the past – and, indeed, has its origins in cognitive science research on human memory and analogical reasoning. CBR involves the tasks of: obtaining a problem description, measuring the similarity of the current problem description against stored problem descriptions, retrieving one or more similar problems from memory along with their solutions, and attempting to reuse a solution - possibly adapting it - to better fit the current problem. This solution, proposed by the system, then gets evaluated, e.g. by being applied to the initial problem, assessed by a domain expert, or tested by other means. The solution is revised, leading to an updated solution; the minimal update being just a confirmation of the proposed solution. The solution can then be retained in memory, for use during future problem solving, and in this sense the system is said to have learned to solve a new problem. Part of the inspiration for the study of case-based reasoning arose from cognitive science, but the resulting methodology has proven useful in an extensive set of applications (e.g., Watson, 1997).

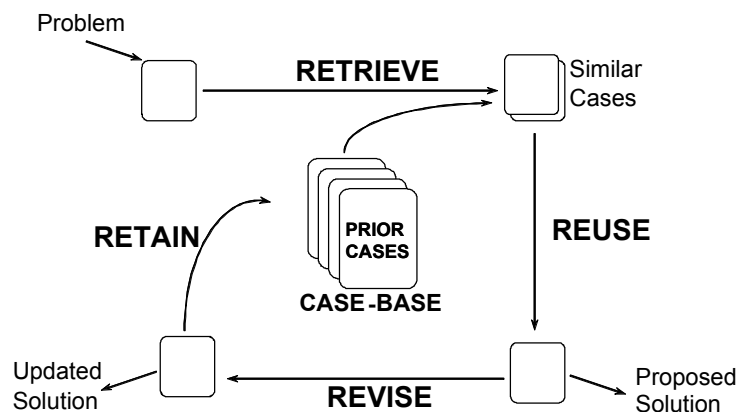


Figure 1 The CBR cycle (Aamodt & Plaza, 1994)

This basic set of CBR tasks was characterized by Aamodt & Plaza (1994) as the CBR-cycle of Figure 1. The individual tasks in the cycle are referred to as the “4 REs”, for retrieve, reuse, revise, and retain. This characterization is still widely used by CBR researchers today, but has itself been embellished as researchers have come to appreciate a broader perspective of case-based reasoning.

Within the CBR-cycle, numerous researchers have concentrated on the retrieval task and the task of measuring similarity, because of the pivotal role of this task in the completion of the cycle. Leake (1996) expressed the role of similarity through the concepts of retrieval and adaptation distances in a simple diagram, shown in Figure 2, which illustrates the relationship in CBR between the problem description and solution description spaces. CBR is based on the observation that similar problems have similar solutions (Leake & Wilson 1999). This observation has been proven to hold in expectation for simple scenarios (Faltings 1997b), and is empirically validated in many domains.

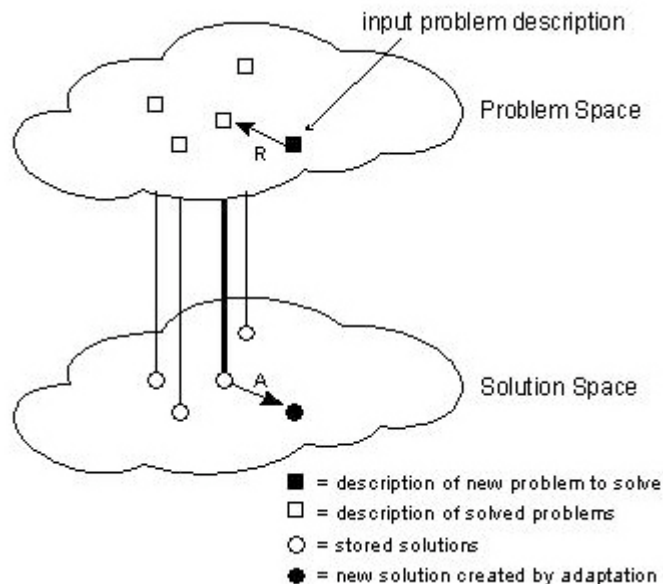


Figure 2 Relationship between problem and solution spaces in CBR (Leake, 1996)

In Figure 2, the retrieval distance R increases as the similarity between the input problem description and a stored problem description decreases (i.e., high similarity means low R). This reflects a common assumption in CBR that the retrieval distance R is commensurate with the adaptation distance (or effort) A . According to this assumption, the validity of which we examine in Section 3, it is easier to adapt the solution of a similar problem than the solution of a less similar problem; although, as we will discuss in section 3, sometimes this assumption cannot be relied upon to hold true, leading to the need for alternative retrieval strategies.

As will be apparent from the literature review that we present in this paper, aspects of reuse and retention, and to a lesser extent revision, have also attracted considerable research interest in CBR. In Section 2, we examine the cognitive science foundations of CBR and the relationship between CBR and analogical reasoning. In Sections 3, 4, and 5, we review a representative selection of CBR research on aspects of case retrieval, reuse, revision, and retention in the past

couple of decades. Our conclusions are presented in Section 6.

2 CBR and cognitive science

The study of CBR has been strongly influenced by research in cognitive science. A major current underlying early CBR research was the study of human story understanding (Schank & Abelson, 1977), especially as it led to investigations of the role of memory in understanding (Schank, 1982). Initial story understanding work examined the knowledge structures underlying understanding and their role in providing expectations for the events in stories. These knowledge structures also provide a causal structure that links the states and events in stories and explains why the agents do what they do. Dynamic memory theory (Schank, 1982) focused on the interplay of understanding, learning, and memory. Memory Organization Packets, or MOPs, organize sequences of events, but individual MOPs may share structure and inherit information from other MOPs. MOPs organize individual events which can be recalled as *reminders*. These reminders can play many roles in interpretation and problem solving. For example, during planning, a problem may prompt the reminding of a past plan that can be adapted to help solve a new problem. This forms a basis for CBR. Reminding may sometimes occur across contexts, enabling the lessons from one situation to be applied to a situation which is superficially quite different.

When expectations fail during understanding, reminders of prior explanations may be useful to help resolve the anomalies present in the input (Schank, 1986; Schank, Kass & Riesbeck, 1994). The SWALE system (Kass, Leake & Owens, 1986; Schank & Leake, 1989), which models case-based explanation generation, uses MOP-based expectations to guide understanding until it encounters an anomaly, and then retrieves prior explanations to adapt to the new situation. The system's namesake was a star race horse whose unexpected death with no warning after a major race, shocked and intrigued the racing community. Experts were immediately reminded of similar cases; one vet's reaction to the news was "This sounds like an aneurysm. I've seen this sort of thing before." The death also prompted less routine reminders among students at the Yale AI lab, such as the death of the rock star Janis Joplin due to an overdose of recreational drugs. While recreational drug use was unlikely for Swale, adaptation to another type of drug use more associated with racing---performance-enhancing drugs---led to a more plausible explanation (though one not borne out by later investigation). SWALE modeled the role of CBR in explanation-building and modeled creativity through methods to perform the flexible retrieval and reuse processes needed to apply explanations in unusual ways.

A number of other studies have explored the role of CBR in human reasoning and learning, giving rise, for example, to teaching systems shaped by lessons from CBR (Schank *et al.*, 1993/1994; Kolodner *et al.*, 2003). A core part of medical diagnostic reasoning is also shown to follow a type of "pattern matching" (Patil, 1986), which in essence is a case-based process of reasoning from past patient experiences. This has given rise to a number of medical CBR systems that support this type of decision making (Bichindaritz, 1995; Ozturk & Aamodt, 1998; Schmidt *et al.*, 2001). More extensive discussions of CBR as a cognitive model are available in (Kolodner, 1994) and (Leake, 1998).

CBR and analogical reasoning

CBR is also fundamentally related to research in analogical reasoning, an active area of research in cognitive science. Analogical reasoning research focuses on basic mechanisms such as matching and retrieval, and how those mechanisms are used in other cognitive processes, including reasoning and learning. Psychological studies have shed light on some basic properties of human analogical reasoning. For example, there is ample evidence that, for people, retrieval is heavily influenced by surface properties more than by deep similarities, unlike most CBR systems, yet when given the analogues, they find the comparisons easy (*cf.* Gick & Holyoak, 1980; Keane, 1988; Gentner, Rattermann & Forbus, 1993). The centrality of relational information in human similarity judgments is now reasonably well-established (*cf.* Gentner, Holyoak & Kokinov, 2001; Markman & Gentner, 2001; Kokinov & French, 2003). This reinforces the practice of using structured representations that was common in early CBR, but it also suggests that many modern CBR and machine learning systems, which use feature-vectors, are unlikely to be good models of human cognition.

Psychological theories of analogical processes have led to computational models that can be used for building CBR systems. For example, the Structure-Mapping Engine SME (Falkenhainer, Forbus & Gentner, 1986; Forbus, Ferguson & Gentner, 1994) is based on Gentner's (1983) structure-mapping theory. SME has been used as a cognitive modeling tool to account for a number of existing psychological findings, and several predictions made based on SME have been subsequently confirmed in psychological experiments (Forbus, 2000). SME has also been used in a variety of performance-oriented systems, ranging from a case-based tutor for engineering thermodynamics (Forbus, 2000) to sketch understanding systems (Forbus, Usher & Chapman, 2003). The IAM system (Keane & Brayshaw, 1988; Keane, Ledgeway & Duff, 1994) was the first to explore incremental mapping of analogies, crucial for extended problem solving. Some computational models have focused on how such computations can be implemented in neural architectures, such as Hummel & Holyoak's (1997) LISA model and Larkey & Love's (2003) CAB model. Other computational models are exploring how statistical association models can be combined with structural models (*cf.* Ramscar & Yarlett, 2003).

One of the major differences in approach between CBR and analogy research is their focus on generality. In analogy work, it is typically assumed that processes like matching and retrieval are broadly general cognitive processes, operating universally (or nearly so) over people's mental representations. In CBR work, where the focus is often on creating a system to do a specific task well on existing computing hardware, generality is often traded in for efficiency or performance, with an emphasis on *content theories* reflecting the knowledge required for particular task domains. Domain-specific matchers, retrieval systems, and even similarity metrics are fair game. This can lead to controversies between the two communities. The dissociation between similarity-based reminding and analogical inference noted above was at first greeted with great skepticism and shock in the CBR community, since it directly contradicted a common assumption that human memory retrieval relied on extensive indexing, using abstract principles. Forbus & Gentner's (1994) MAC/FAC model captures this dissociation by postulating a first stage of retrieval that is non-structural, a cheap filter that generally lets highly similar items through, followed by a much more constrained structural match stage using SME. If the MAC/FAC model is correct, then CBR index-based retrieval schemes would best be viewed as

good engineering tools, rather than as cognitive models. This is still an open question.

CBR also differs from analogy research in its treatment of adaptation. Case adaptation is a major issue for CBR, and must be addressed to develop practical CBR systems. Consequently, it is possible that the experience of CBR in this area will suggest fruitful questions to consider in the context of analogy. Studies focusing on the integration of adaptability concerns into other parts of the CBR process include Smyth & Keane's (1998) adaptation-guided retrieval and Leake, Kinley & Wilson's (1995) use of adaptation cases to reuse prior adaptations and predict adaptability.

Discussion

How closely should CBR systems mirror what people do? It can be argued that, for building practical CBR systems, deliberately organizing them to operate differently to people in some ways might make them more useful. Just as eyeglasses and cars help us see and move farther than we can unaided, carefully designed CBR systems could help us retrieve more relevant memories more often, and help us to work through problems that we could not unaided due to working memory limitations. On the other hand, no CBR system comes within several orders of magnitude of the amount of knowledge and experience that people accumulate, nor has one CBR system ever operated over the breadth of kinds of problems that people can handle. The processes of human cognition may well hold the secrets to creating such capabilities.

3 Retrieval in CBR

An important step in the CBR cycle shown in Figure 1 (Aamodt & Plaza, 1994) is the retrieval of previous cases that can be used to solve the target problem. The so-called *similarity assumption* tells us that when it comes to selecting such past cases, we should look to those that are most similar to the target problem. As will be apparent from our review of selected papers from the CBR literature on aspects of similarity-based retrieval, this assumption remains deeply ingrained in CBR research.

The success of similarity-based retrieval approaches relies on an accurate assessment of case similarity to the target problem. Consequently, considerable research has been devoted to improving retrieval performance through the development of effective approaches to similarity assessment. However, in recent times the similarity assumption has been brought into question in a number of ways, leading to an improved understanding of the limitations of similarity-based retrieval, especially in light of new application requirements. For instance, Smyth & McClave (2001) point out that selecting cases by their similarity to the target problem may result in the retrieval of a set of cases that provide relatively poor coverage of the problem space in the vicinity of the target. Their observation that pure similarity-based retrieval often results in the selection of cases that are lacking in *diversity* has prompted the development of a range of alternative retrieval methods for selecting more *representative* sets of cases (e.g., Smyth & McClave, 2001; McSherry, 2003c); see also section 3.3.

Similarly, different sources of retrieval knowledge have been proposed as a way to better coordinate the retrieval and reuse stages of CBR. Sometimes, for example, the most similar case to a target problem is not the easiest to adapt. At best this can reduce adaptation efficiency but it may even lead to a reuse failure, if a retrieved case cannot be successfully adapted. In response to

this researchers have considered the adaptability of cases as a guide for case retrieval (Smyth & Keane, 1995a; 1998; Leake, Kinley & Wilson, 1997). Moreover, in interactive case-based reasoners, where case adaptation may be less of an issue, but where the case-based reasoner must be in a position to *explain* or *justify* the reasons behind a retrieval, it is appropriate to consider the explanatory potential of cases. For instance, early work on case-based legal argumentation (Ashley, 1989; 1991; Ashley & Aleven, 1997) broadens the notion of case similarity to include other considerations (e.g., noteworthy distinctions among cases, the existence of counterexamples) with a view to emphasizing the roles of explanation and argumentation in CBR systems. Recent work by Doyle *et al.* (2004) further develops this theme by evaluating the explanatory potential of cases as a retrieval constraint.

3.1 Similarity assessment

In some applications of CBR, it may be adequate to assess the similarity of the stored cases in terms of their *surface* features. The surface features of a case are those that are provided as part of its description and are typically represented using attribute-value pairs. In other applications, it may be necessary to use *derived* features obtained from the case description by inference based on domain knowledge. Note that although the following discussion will generally treat “surface features” as relatively superficial, this is a simplification for convenience. As Kolodner (1996) observes, strictly speaking, whether a feature is predictive of a case’s relevance, or whether it is superficial or deep, is orthogonal to whether the feature is readily available. In yet other applications, cases are represented by complex structures (such as graphs or first-order terms) and retrieval requires an assessment of their *structural* similarity. As might be expected, the computation of deep features or use of structural similarity is computationally expensive; however, the advantage is that more relevant cases may be retrieved.

One way to help assure useful retrievals, without the need for extensive computation, is to develop carefully crafted *indexing vocabularies* to describe cases, so that the explicit description of a case captures the features which determine its relevance. Thus, the focus of considerable early CBR work concerned the development of such indexing vocabularies for a number of domains (Schank *et al.*, 1990; Domeshek, 1992; Leake, 1992) as a way to avoid the need for computationally expensive structure mapping and case matching procedures.

Assessment of surface similarity

In approaches to retrieval based on surface features, the similarity of each case to the target problem, typically represented as a real number in the range from 0 to 1, is computed according to a given similarity measure. Usually the retrieved cases are the k cases that are most similar to the target problem, an approach often referred to as “ k nearest neighbor” retrieval or simply k -NN. Alternatively, the retrieved cases may be those whose similarity to the target problem exceeds a predefined threshold.

There are many ways of measuring similarity and different approaches are appropriate for different case representations. For example, it is common in practice for each case to be represented as a simple feature vector (or set of attribute-value pairs). With this representation, it is usual to define a *local* similarity measure for each attribute. A *global* similarity measure aggregates the degrees of similarity that are computed by the local measures. This is most easily

done when the degrees of local similarity are represented by real numbers normalized to some fixed interval such as $[0,1]$. Then, global similarity can be computed as a weighted average of the local similarities. The weights allow different attributes to have different importance; they may be assigned by a domain expert or a user, or as discussed in Section 3.2, they may be determined by an adaptive learning process.

A CBR system can guarantee that it retrieves the k cases that are maximally similar to the target problem by computing the similarity of the target problem to every case in memory. However, this process has complexity $O(n)$ where n is the number of cases in memory, which may be an unacceptable overhead if n is very large. One approach to reducing retrieval time, as exemplified by the pioneering work of Stanfill & Waltz (1986), involves the use of massively parallel computers. While the requirement for expensive hardware is an obvious drawback, the approach still guarantees to find the maximally similar cases by performing an exhaustive memory search. Stanfill & Waltz describe the implementation of a memory-based reasoning algorithm on a fine-grained SIMD parallel machine. Their *Connection Machine* performs a highly parallel search for similar cases and has been applied to the problem of pronouncing English words using a case memory containing thousands of examples of words that are correctly pronounced.

Another approach to reducing retrieval time relies on the organization of cases in memory. For example, Wess, Althoff & Derwand (1993) propose an approach to retrieval in which the organization of the case memory is based on similarities *between* cases. A binary tree called a k - d tree is used to split the case base into groups of cases in such a way that each group contains cases that are similar to each other according to a given similarity measure. To ensure that the most similar cases are retrieved, the retrieval algorithm computes similarity bounds to determine which groups of cases should be considered first.

Smyth & McKenna (1999a; 2001b) propose an alternative model of case retrieval that is informed by the availability of an explicit model of case-base competence (Smyth & McKenna, 1998; 2001a). The so-called *footprint-based retrieval* algorithm is a two-stage retrieval approach that searches two distinct populations of cases. First, it involves the search of a small subset of so-called *footprint* cases, which have been identified as providing a covering set for the case base as a whole. That is, the footprint cases are predicted to be able to solve the same set of problems as the case base as a whole. They are drawn from the key competence groups that exist within the case base as made available by the competence model. The first stage of retrieval identifies the footprint case that is most similar to the target problem as the *reference* case, and the second stage of retrieval then searches another small subset of cases that are related to this reference case. This *related set* is chosen because its cases either cover (solve) the reference case or because they can be covered by a reference case. The final case chosen for retrieval is the related set case that is most similar to the target problem. The approach has been shown to offer significant retrieval efficiency benefits, by searching only a small fraction of the cases in the case base, while at the same time guaranteeing the selection of near optimal cases. Footprint-based retrieval is somewhat related to the “Fish-and-Shrink” strategy (Schaaf, 1996) in which cases are linked according to specific aspect similarities. The latter approach relies on the assumption that if a case does not fit a query then this will reduce the likely usefulness of its neighbors. This allows for the efficient elimination of many cases during retrieval.

Simoudis & Miller (1990) argue that retrieval based only on surface similarity may not be sufficiently discriminating when applied to large case memories, and needs to be combined with other techniques in order to reduce the number of cases to be considered for adaptation. They present an approach called *validated retrieval* that is capable of dramatically reducing the number of potentially relevant cases. Retrieval based on surface similarity is combined in the approach with validation of the retrieved cases to determine if they are applicable to the target problem. Associated with each case in memory is a *validation procedure* consisting of a set of domain-specific tests and their results for that case. In order to validate a retrieved case, the tests associated with the case are applied to the target problem. The retrieved case is considered to be relevant to the target problem only if all the tests give the same results for the target problem as they do for the retrieved case.

The *validation* phase that follows the initial retrieval of cases in validated retrieval resembles the *justification* phase in CASEY, a CBR system for medical diagnosis (Koton, 1988). The goal of CASEY's justification component is to determine whether the causal explanation of a retrieved case applies to the target problem. Often this enables CASEY to avoid invoking its causal model when creating an explanation for the target problem. Other systems that combine retrieval based on surface similarity with an additional filter to improve retrieval performance include CHEF (Hammond, 1986), SWALE (Kass, Leake & Owens, 1986), KRITIK (Goel & Chandrasekaran, 1989), and PROTOS (Porter, Bareiss & Holte, 1990).

Assessment of structural similarity

Retrieval based on structural similarity is computationally expensive because it makes extensive use of domain knowledge but, as previously mentioned, the advantage is that more relevant cases may be retrieved. One way of mitigating the extra cost is to use a two-stage retrieval algorithm. Forbus, Gentner & Law (1994) present a computational model called MAC/FAC in which surface and structural similarity are combined in a two-stage process. The first stage uses a surface matcher to filter out a set of potentially relevant items from long-term memory (those that are most similar to the target problem). The selected candidates are processed in the second stage by a *structure-mapping engine* that computes structural matches between them and the target problem and selects one or more of the best matches. Experiments based on human assessment of similarities and analogies have confirmed that both surface and structural similarity assessment are necessary for sound retrieval (Forbus *et al.*, 1994). Inspired by previous work by Gentner & Forbus (1991), Börner (1993) proposes an approach to retrieval in which fast retrieval of candidate cases based on their surface similarity to the target problem is followed by a more expensive assessment of their structural similarity. She defines structural similarity as the most specific graph structure that the target problem has in common with a stored case, and a set of transformation rules, given as background knowledge, needed to determine this common structure.

Object-oriented case representations generalize simple attribute-value representations. Cases are represented by sets of objects. Objects belong to classes, which are organized in a class hierarchy. An object's class determines the attributes it may have. Attributes may be *relational*, which means that their values will themselves be further objects. It seems obvious that the class hierarchy must contain useful similarity knowledge. For example, objects that are close to each

other in the hierarchy are likely to be more similar than objects that are far apart. However, Bergmann & Stahl (1998) suggest that because there is no clear view about how the similarity between objects of different classes should be determined, the assessment of similarity is often restricted to objects of the same class. To address this issue, they present a new framework for computing object similarities for object-oriented case representations that enables objects of different classes to be compared and takes account of the knowledge that is implicit in the class hierarchy.

In the Creek system (Aamodt, 1994; 2004) an object-oriented, frame-based representation system is used to capture both cases and general domain knowledge, which together can be viewed as one single, multi-relational semantic network. Similarity assessment is once again a two-step process in which the use of direct indexes to retrieve a set of potentially similar cases is followed by a closer examination of the cases in which general domain knowledge is utilized to generate explanations for feature-to-feature matches. A method inspired by Cohen's (1985) work on endorsement theory and plausible inference constitutes a core part of the inference machinery underlying the generation and evaluation of explanatory structures.

Spreading activation methods (eg., Brown, 1994) represent case memory as an interconnected network of nodes capturing case attribute-value combinations. Activation spreads from target attribute-value nodes across the network to cause the activation of case nodes representing similar cases to the target. The approaches are efficient and flexible enough to handle incomplete case descriptions, but there can be a significant knowledge-engineering cost associated with constructing the activation network. Furthermore the spreading-activation algorithm requires specific knowledge to guide the spread of activation throughout the network. Related network-based retrieval methods are proposed by Wolverton & Hayes-Roth (1994) and Lenz (1996).

Another way of dealing with relations between attributes is based on the concept of *generalized cases* (Bergmann *et al.*, 1999). A generalized case covers a subspace of the problem-solution space, providing solutions to a set of closely-related problems, rather than just a single problem (Mougouie & Bergmann, 2002). Dependencies between attributes are explicitly represented in a way that supports the extension of similarity measures. For example, Bergmann (2002) defines the similarity between a query and a generalized case as the similarity between the query and the most similar case contained in the generalized case. Mougouie & Bergmann (2002) formulate the similarity assessment problem for generalized cases, described by continuous attributes, as a nonlinear programming problem and introduce an optimization-based retrieval method. Tartakovski *et al.* (2004) extend the case representation to support mixed, discrete, and continuous attributes. They also formulate similarity assessment as a special case of a mixed integer nonlinear optimization problem, and propose an optimization-based retrieval method operating on a given index structure.

Bunke & Messmer (1993) propose one of a number of structural similarity measures for domains in which cases are represented as graph structures. Their proposed measure is based on *graph editing* operations (inserting, deleting, and substituting nodes and edges in the graph). To improve the practical efficiency of the approach, they introduce a *subgraph matching* algorithm that works on a compactification of the case memory in which subgraphs which are common to multiple cases may be stored only once. In a similar vein, Champin & Solnon (2003) propose a similarity measure, based on graph editing operations within a modification of Tversky's (1977)

contrast model, to compare cases represented by labeled graphs where vertices and edges can have more than one label. To deal with the intractability of this representation, a heuristic greedy algorithm is proposed.

Arcos & López de Mántaras (1997) describe a retrieval mechanism called *Perspectives* for structured case representations. Cases and degrees of similarity are represented as feature terms, which are equivalent to first-order terms and can also be viewed as directed acyclic graphs labeled by features and values (Plaza, 1995). Their knowledge-intensive approach to retrieval uses a *subsumption* mechanism between the feature terms to obtain an order relation between case descriptions on the basis of a set of user-defined relevant aspects of the target problem. The system is implemented in an object-oriented language (Arcos, 1997) based on feature terms and has been applied to the problem of synthesizing expressive music (Arcos & López de Mántaras, 2001; López de Mántaras & Arcos, 2002).

Emde & Wettschereck (1996) propose an alternative way of measuring the similarity of first-order terms. They also present a generalization of a propositional instance-based learner (distance-weighted k -NN) to first-order representations. Issues addressed in the approach, which the authors refer to as *relational* instance-based learning (RIBL), include the generation of cases from the knowledge base, assessment of similarity between arbitrarily complex cases, and estimation of the relevance of predicates and attributes. Empirical results are presented which suggest that RIBL is capable of achieving high levels of classification accuracy in a variety of domains.

Similarity frameworks

With so many ways of measuring similarity, it is unsurprising that some researchers have looked at similarity in a general way, independent of any specific algorithm. For example, Richter (1992) discusses the notion of similarity in the context of a formal mathematical framework. He describes approaches to modeling similarities with increasing complexity and informativeness. These range from simple predicates (least informative) to relations and functions (most informative) and general forms of distance functions and similarity measures are discussed, including a generalization of Tversky's (1977) *contrast model*. The contrast model is based on a set-theoretic approach that expresses the similarity between objects as a linear combination of their numbers of matching and mismatching features. One limitation of Tversky's model is that all features are assumed to be equally important, whereas Richter's generalization allows different weights to be assigned to features. However, Richter emphasizes that to allow for changes in the problem-solving environment, the parameters of a similarity measure should be the result of an adaptive learning process, an idea we explore further in Section 3.2.

Osborne & Bridge (1996) present another general framework distinguishing, in particular, between *ordinal* and *cardinal* similarity measures. Ordinal measures use a description of the target problem to induce a partial ordering over the cases in the case memory. No information about the degree of similarity is given; the cases are merely ordered, with the implication that cases higher in the ordering should be retrieved prior to any that are lower in the ordering. Cardinal measures, on the other hand, are functions that score the cases, returning real numbers to denote degrees of similarity. Osborne & Bridge present a set of operators that allows the flexible and systematic construction of both ordinal and cardinal similarity measures. In later

work (e.g., Osborne & Bridge, 1997), the framework is generalized further to accommodate not only similarity measures in which the degree of similarity is denoted numerically (which is most common) but also similarity measures in which the degree of similarity is instead denoted by any value drawn from an ordered set. With this extension, the framework accounts for similarity measures in which the degree of similarity is denoted by common subgraphs (Börner, 1993) or feature terms (e.g., Jantke, 1994; Plaza, 1995).

3.2 *Improving and evaluating retrieval performance*

Several techniques for improving the *speed* of retrieval were mentioned in our discussion of similarity assessment in Section 3.1. Another important aspect of retrieval performance is its impact on *solution quality*. Measures used to evaluate retrieval performance in terms of solution quality are likely to depend on the type of problem-solving task (e.g., classification, recommendation, planning) for which the system is designed. For example, evaluation in terms of *classification accuracy* is possible only if the outcome classes to be predicted in the *test set* are represented in the *training set*. This is not the case in domains such as product recommendation in which each outcome class (a unique product or service) is represented by a single case in the case memory (McSherry, 2001a). Evaluation of retrieval performance in terms of classification accuracy is similarly compromised in *conversational CBR* (Aha *et al.*, 2001), where it is typical for most cases to have unique solutions. Appropriate measures of retrieval performance for datasets of this type include precision, recall, and the average length of problem-solving dialogues (Aha *et al.* 2001; McSherry, 2001a; 2001b; 2003b; McGinty and Smyth, 2003).

Problems likely to affect retrieval performance in terms of solution quality include the use of inadequate similarity measures, missing values in cases, unknown values in the description of the target problem, and the so-called *heterogeneity* problem that arises when different attributes are used to describe different cases (Aha *et al.*, 2001; McSherry, 2001b; 2003c; Stahl & Gabel, 2003; Bogaerts & Leake, 2004). Bogaerts & Leake (2004) propose and evaluate a variety of possible strategies for handling missing information in similarity assessment. Retrieval based on incomplete information is an important challenge in conversational CBR, where a description of the target problem is incrementally (and often incompletely) elicited in an interactive dialogue with the user. Aha *et al.* (2001) evaluate an approach to incremental query elicitation that takes account of the heterogeneity that is typically found in domains such as fault diagnosis. McSherry (2003b) proposes a conversational CBR approach to product recommendation that includes a mechanism for ensuring that the dialogue is terminated only when it is certain that a more similar case will not be found if the dialogue is allowed to continue.

Retrieval performance can often be improved by making the similarity measure the subject of an adaptive learning process. Focusing on variants of *k*-NN that automatically learn the weights assigned to features, Wettschereck & Aha (1995) propose a multi-dimensional framework for the categorization and comparison of feature weighting methods in CBR. The proposed framework can be used to categorize new methods, thus facilitating their comparison with existing methods. However, it cannot be applied to *k*-NN methods that incorporate domain-specific knowledge and complex representations. Noting that most feature weighting methods are designed to optimize classification accuracy, Wilke & Bergmann (1996) argue that decision *costs* should also be considered in many applications. Experimental results are presented which support the

hypothesis that classification based on weights learned using cost optimization leads to lower decision costs than classification based on weights learned by accuracy optimization.

Improving the adaptability of retrieved cases can also be the subject of an adaptive learning process. In case-based planning, for example, Muñoz-Avila & Hüllen (1996) extend the *foot-printed* similarity metric used in PRODIGY/ANALOGY (Velo, 1992) by incorporating feature weights in a new metric which counts the weights of relevant features that match features in the target problem. A feature is considered relevant to a planning goal with respect to a solution if it contributes to achieving the goal in the solution. The authors also present an algorithm for analyzing the performance of retrieved cases to identify features whose weights need to be recomputed. The algorithm provides a bridge between the new similarity metric and a feature weighting model based on incremental optimizers. Experimental results are presented which show that integration of the proposed similarity metric and analysis algorithm in the feature weighting model improves the adaptability of the retrieved cases by convergence to best weights over a period of multiple problem-solving episodes.

Many CBR applications rely on domain knowledge encoded in the similarity measures used by the system to guide the retrieval of relevant cases. Such a *knowledge-intensive* approach to similarity assessment typically relies on knowledge acquired from a domain expert. In a recent series of papers, Stahl & Gabel investigate the use of machine learning techniques to reduce the knowledge-acquisition overheads associated with the construction and maintenance of domain-specific similarity measures (Stahl & Gabel, 2003; Gabel & Stahl, 2004). A distinctive feature of their approach is the use of feedback about the quality of retrieval results provided by a domain expert to guide the automatic refinement of similarity measures.

3.3 Alternatives to similarity-based retrieval

Much of the research on which we comment in this section has been motivated by an increased awareness of the limitations of retrieval based purely on similarity. While continuing to play a prominent role in retrieval, similarity is increasingly being combined with other criteria to guide the retrieval process, such as how effectively the solution space is *covered* by the retrieved cases (McSherry, 2003c), how easily their solutions can be *adapted* to solve the target problem (Smyth & Keane, 1998), or how easily the proposed solution can be *explained* (Doyle *et al.*, 2004).

Adaptation-guided retrieval

While many factors may contribute to the performance of the retrieval component of a CBR system, in many applications what matters the most is whether the retrieved cases can be used to solve the target problem. The issue is not finding cases that are similar *per se*; it is a question of finding cases that are *usefully* similar. This view is perhaps most formally expressed by Bergmann *et al.* (2001), who explicitly state that problem similarity is used as a proxy for solution *utility* – the similarity assumption rephrased. But the problem remains that similarity may be inadequate as a proxy for solution utility and this raises the question of how best to handle such an eventuality. The solution, as it turns out, is based on the idea that there are forms of knowledge other than case similarity that can, and should, be brought to bear on the retrieval task.

For example, Smyth & Keane (1994; 1995a; 1996; 1998) question the core similarity assumption according to which the most similar case is the one that is easiest to adapt. They argue that sometimes the most similar case may be impossible to adapt, for example if adaptation knowledge is incomplete, as is often the case in weak-theory domains that are commonly targeted by CBR. To address this issue, they introduce the notion of *adaptation-guided retrieval* in which the adaptation requirements of cases are explicitly assessed during retrieval by means of domain-specific adaptation knowledge. In contrast to traditional approaches that relied on heuristics to predict the ease with which a given case could be adapted, adaptation-guided retrieval combines explicit local and global measures of adaptability to ensure that the most adaptable case is always selected. In this way adaptation-guided retrieval bridges the gap between retrieval (and its similarity knowledge) and reuse (and its adaptation knowledge). Empirical results demonstrate how the approach can lead to a significant reduction in adaptation failures as well as a concomitant decrease in adaptation costs by performing preliminary adaptation work during retrieval. Leake, Kinley & Wilson (1997) propose a case-based approach to this problem, predicting adaptation effort based on prior adaptation experiences.

Diversity-conscious retrieval

Recently there has been considerable interest in interactive case-based reasoners, which avoid the problems associated with automating reuse by leaving adaptation to the user. They focus instead on the retrieval of high-quality cases. A good example is provided by CBR approaches to product recommendation, in which the role of the CBR system is to select cases for recommendation based on a query describing the user's requirements. If there is no case that exactly matches the user's requirements, she can be shown the cases that are *most similar* to her query. A limitation of similarity-based retrieval, especially in a recommendation context, is that the most similar cases are often very similar to each other (Smyth & McClave, 2001). In other words, the top cases often lack diversity and offer the user a very limited choice of alternatives. Recently there has been considerable research interest in algorithms that combine measures of similarity and *diversity* in the retrieval process to achieve a better balance between these often conflicting characteristics (e.g., Smyth & McClave, 2001; McSherry, 2002; McGinty & Smyth, 2003). For example, Smyth & McClave (2001) propose an approach to retrieval that incrementally selects a set of diverse cases from a larger set of similarity-ordered cases. Experimental results have shown that major gains in diversity can often be achieved at the expense of relatively small reductions in similarity. As a consequence, users receive suggestions that better cover the relevant region of the target product space, increasing the likelihood that they will be recommended a satisfactory product.

Compromise-driven retrieval

McSherry (2003c) proposes a *compromise-driven* approach to retrieval in recommender systems inspired by the observation that the cases that are most similar to the user's query are often not sufficiently representative of compromises (or unsatisfied requirements) that the user may be prepared to accept. An underlying assumption of similarity-based retrieval (or *k*-NN) is that a given case (or product) is more acceptable than another if it is more similar to the user's query. Compromise-driven retrieval is based on the weaker assumption that a given case is more

acceptable than another if it is more similar to the user's query *and* it involves a subset of the compromises that the other case involves. As well as being less likely to be contradicted by user behavior, this weaker assumption provides the basis of a more principled approach to deciding which cases are included in the retrieval set than arbitrarily limiting the number of retrieved cases as in k -NN. For example, no case is included in the retrieval set if there is a more similar case that involves a subset of the compromises it involves.

Though not relying on an explicit *measure* of diversity in the retrieval process, compromise-driven retrieval shares with other approaches to enhancing recommendation diversity (Smyth & McClave, 2001; McSherry, 2002) the aim of offering users a better choice of alternatives. Another important benefit of the approach is that the retrieval set is guaranteed to provide full *coverage* of the available cases in the sense that for any case that is not included in the retrieval set, one of the recommended cases is at least as good in terms of its similarity to the user's query and the compromises it involves.

Order-based retrieval

Order-based retrieval is another new approach with particular application to recommender systems (Bridge & Ferguson, 2002a). Rather than scoring the cases, order-based retrieval offers an expressive query language for defining and combining ordering relations; the result of query evaluation is to partially order the cases in the case base. The claims made for the approach include: it is more expressive than similarity-based retrieval because it allows queries that naturally combine not just the user's preferred value (one for which similar values are sought) but also dispreferred values, minimum values, maximum values, and so on; and it returns inherently diverse result sets (Bridge & Ferguson, 2002b).

Explanation-oriented retrieval

It is often important for CBR systems to be able to explain their reasoning (e.g., Rissland, *et al.* 1984; Ashley & Alevan, 1992; Leake, 1996; McSherry, 2001d; Cunningham *et al.*, 2003; Doyle *et al.*, 2004; McSherry, 2004a) and to justify their suggestions or solutions. Explanations serve many different *goals*, such as teaching the user about the domain or explaining the relevance of a question the user is asked (Leake, 1991; 1992; Sørmo & Cassens, 2004; Sørmo, Cassens & Aamodt, 2005/2006). For example, McSherry (2003b; 2004b) proposes a conversational CBR approach to product recommendation in which the system can explain why a question is being asked in terms of its ability to discriminate between competing cases. Explaining the *retrieval failures* that occur when no case exactly matches the user's query is another potential role of explanation in CBR recommender systems (e.g., McSherry, 2003a).

More commonly, the goal is to explain how the system reached its conclusions. In applications such as classification and diagnosis, an attractive feature of CBR is the ability to explain the predicted outcome by showing the user one or more of the target problem's nearest neighbors. As noted by Leake (1996), "... *the results of CBR systems are based on actual prior cases that can be presented to the user to provide compelling support for the system's conclusions*". Such explanations are known as precedent-based explanations and have long been a feature of case-based models of legal argumentation (e.g., Ashley, 1991; Branting, 1991; Rissland & Skalak, 1991). An empirical study by Cunningham *et al.* (2003) has shown that they

are often more compelling than alternative forms of explanation. However, a number of authors have recently questioned the effectiveness of precedent-based explanations in which the user is simply shown the case that is most similar to the target problem.

For example, McSherry (2004a) argues that such explanations are often less informative than might be expected, and should ideally be supported by an analysis of the *pros* and *cons* of the proposed solution. Doyle *et al.* (2004) have found that the most compelling explanation case may not necessarily be the one that is most similar to the target problem. In particular, they demonstrate how cases that lie between the target problem and the decision boundary can often be more useful for explanation. This has motivated the development of *explanation-oriented* retrieval. This approach remains precedent-based, but once once a classification or diagnosis has been reached on the basis of the nearest neighbors, the system performs an additional retrieval step, using an explanation utility metric, to obtain the explanation case. Doyle *et al.* (2004) also report the results of an empirical study that show their explanation cases to be generally more compelling than the nearest neighbor.

Other important aspects of case-based explanations that have been explored in work on case-based legal argument (Ashley, 1989, 1991; Ashley & Alevén, 1992; 1997; Alevén, 2003) include explaining and distinguishing relevant cases' strengths and weaknesses, providing examples to resolve conflicts, presenting counterexamples to proposed solutions, making *ceteris paribus* comparisons, and posing hypothetical variations of problems to illustrate their effects on the analyses.

4 Reuse and revision in CBR

The reuse process in the CBR cycle is responsible for proposing a solution for a new problem from the solutions in the retrieved cases. In Aamodt & Plaza's (1994) "four REs" cycle (Figure 1), ReUse appears second, after ReTrieve, and is followed by ReVise and ReTain. Reusing a retrieved case can be as easy as simply returning the retrieved solution, unchanged, as the proposed solution for the new problem. This is often appropriate for classification problems such as interpretation, diagnosis, monitoring and prediction. For classification tasks, each solution (class) is likely to be represented frequently in the case base, and therefore the most similar retrieved case is likely to contain an appropriate solution. But reuse becomes more difficult if there are significant differences between the new problem and the retrieved case. In these circumstances the retrieved solution may need to be adapted to take account of these important differences. Medical decision making is one domain in which adaptation is commonly required.

Adaptation becomes particularly relevant when CBR is used for constructive problem-solving tasks such as design, configuration, planning. For such tasks it is unlikely that each solution (design, configuration, or plan) will be represented in the case base. Thus the retrieved solution is simply an initial solution and any differences between the new problem (specification) and the retrieved case are likely to alter the retrieved solution in some way.

Adaptation methods differ in complexity with respect to two dimensions: what is changed in the retrieved solution, and how the change is achieved. *Substitution* methods simply reinstantiate some part(s) of the retrieved solution, whereas *Transformation* methods alter the structure of the solution (Kolodner, 1993). Adaptation is commonly achieved by altering the retrieved solution directly, but the more complex *Generative Adaptation* replays the method of deriving the

retrieved solution on the new problem. These three types of adaptation methods will be used to structure this section. The contributions discussed in this section describe different approaches to adaptation for reuse; i.e. adaptation during solution formulation. Adaptation can also be used when feedback about a proposed solution indicates that a repair is needed; this is part of the ReVise stage in the CBR cycle.

Hammond (1990) describes the reuse of recipes in CHEF, a menu-planning system. Substitution Adaptation is used to substitute ingredients in the retrieved recipe to match the menu requirements; e.g., when a recipe containing beef and broccoli is retrieved for a meal requiring chicken and snow peas, then the meat component is replaced by chicken and the vegetable component is substituted by snow peas. Transformational adaptation may also be needed to amend the proposed recipe further by adding or removing steps in the recipe that result from any ingredient substitutions; e.g., for chicken, rather than beef, a new skinning step should be added. Further transformations may occur at the ReVise stage where critics analyze the failure of a recipe and repair strategies are applied to the proposed recipe to add or remove steps in the failed recipe. CHEF's learning of critics introduced the topic of case-based planning and many of its themes; e.g., indexing, use of cases in memory, failure-driven learning.

SWALE (Schank & Leake, 1989) is a case-based explanation system for story understanding which reuses old explanations by applying substitution adaptation to amend the actor, their role or the action in the retrieved explanation (Kass, 1989). Again transformational adaptation may be needed to add or remove components in the current explanation resulting from these substitutions.

Déjà Vu (Smyth & Keane, 1995a; 1998) is a CBR system for the automated design of plant-control software that builds on some of the ideas proposed in CHEF (Hammond, 1990) by utilizing transformational adaptation knowledge in the form of general *adaptation strategies* and more specialized *adaptation specialists*. An important and novel contribution of Déjà Vu is its representation of complex plant-control software designs as hierarchies of related cases and its adoption of a hierarchical model of case retrieval and reuse. For example, complex target problems lead to the retrieval and adaptation of the solutions to *abstract cases*, the elements of which in turn lead to the retrieval and adaptation of more detailed sub-cases. In this way, the challenge of solution transformation is answered by a combination of problem decomposition, sub-case adaptation, and solution re-integration. Moreover, a unique feature of Déjà Vu, as discussed in Section 3.2, concerns its ability to leverage existing adaptation knowledge during retrieval in order to evaluate the adaptability of cases. This further helps to alleviate some of the adaptation problems by at least helping to guarantee the retrieval of cases that can be adapted easily.

Model-based adaptation is a popular approach to transformational adaptation in which causal reasoning is integrated with CBR. Koton's (1988) CASEY is an early example of model-based adaptation, in a medical diagnosis CBR system that utilizes domain independent repair strategies to adapt the retrieved explanation to take account of differences between the symptoms of new and retrieved patients. KRITIK (Goel & Chandrasekaran, 1989) relies on model-based transformation adaptation to reuse designs for physical devices. Model-based reasoning creates a causal explanation for the new design by transforming that of the retrieved one. Faltings' (1997a) CADRE also applies model-based reasoning in case-based design. Reuse involves the

combination of retrieved design cases and the transformation adaptation of the retrieved design. This work is evaluated in two design prototypes: CADRE for architectural design and FAMING (Faltings & Sun, 1996) for mechanism design.

Evolutionary methods have also been explored for adaptation, in the context of architectural design (Gómez de Silva Garza & Maher, 2000). The retrieved designs become the initial population for a *Genetic Algorithm* and mutation and crossover operators are used to generate new designs for the population. Mutation is a substitution adaptation that randomly alters parts of one design to produce a new design. Crossover is a transformation adaptation that can alter the structure of the design. It generates two new designs from two parent designs by interchanging parts of the design in each parent. The genetic algorithm's fitness function evaluates the designs by calculating how well they match the design requirements. The design with the highest match to the requirements is the new design. Purvis & Pu (1995) present adaptation as a *Constraint Satisfaction Problem*. The design cases are represented as constraint satisfaction problems, where the design requirements are the constraints and the design is the solution. The retrieved designs are adapted by applying a minimum conflicts heuristic to guide the repair of the design to match the new design requirements.

The work discussed so far in this section has been devoted to substitution and transformation adaptation. Based on studies of several CBR systems that use adaptation, Fuchs & Mille (1999) have proposed a knowledge-level task model for substitution and transformation adaptation processes. The ReUse task is composed of *Copy* and *Adapt* tasks. The Adapt task comprises selecting a problem difference, *Modifying* the solution, and finally verifying the solution. The Modify task can remove or *substitute/add* part of the solution, and finally the substitute/add task searches for a suitable replacement by using further cases, applying a heuristic, or accessing domain knowledge such as explanations, abstractions or specializations.

Generative Adaptation differs from substitution and transformation adaptation in that it does not adapt the retrieved solution directly, but instead derives the new solution by replaying the method used to derive the retrieved solution. Generative adaptation may result in a reinstatement of parts of the retrieved solution, like substitution, or in a transformation that alters the structure of the solution. PRODIGY (Velooso & Carbonell, 1994), a general purpose planning system, applies *Derivational Replay* to recompute a replacement for a faulty element of the retrieved solution by recalling how the element was computed and replaying the computation for the new problem. Derivational Replay is a variant of *Derivational Analogy* (Carbonell, 1986) in which the complete solution is recomputed. In this work an analogy between the new and retrieved problems is used to adapt the method of deriving the solution.

Although CBR systems avoid reasoning from first principles by remembering and reusing past solutions, substitution and transformation adaptation of retrieved solutions is often achieved by reasoning about how the problem differences should be reflected in the adaptation to the proposed solution. Therefore the acquisition of adaptation knowledge can require a substantial knowledge engineering effort. The difficulty of acquiring adaptation knowledge was identified in early CBR research but, until recently, relatively little effort has been devoted to automating the acquisition of adaptation knowledge. Leake, Kinley & Wilson's (1995) DIAL system for disaster response planning builds up its adaptation knowledge as it applies case-based planning. The adaptation knowledge it learns is a set of adaptation cases that capture the steps in successful

manual plan adaptations. DIAL applies a mixed-initiative adaptation process: if an adaptation case matches the current adaptation need it is reused by a CBR process, otherwise DIAL attempts to apply a general rule to revise the plan (e.g., add or remove a step), but as a final option it resorts to manual adaptation. This last option offers the opportunity to acquire a new adaptation case when the manual adaptation generates a successful plan.

Several systems exploit the knowledge already captured in the cases as a source of adaptation knowledge. McSherry (1998) reuses pairs of cases from the case base that contain the same differences as those found between the new problem and the retrieved case. The solution difference from the pair of cases is replayed on the retrieved case. Rather than reusing differences directly from the cases, Hanney & Keane (1997) use the case-base as a source of case pairs that are used as training data to learn rule-based adaptation knowledge that generalizes the adaptations represented in the case pairs. Wilke, Vollrath, Althoff & Bergmann (1997) present a framework for *knowledge-light* learning of adaptation knowledge as in these approaches, where knowledge contained elsewhere in the CBR system, like the case base, is used to learn or improve the adaptation knowledge. The final paper discussed in this section (Jarmulak, Craw & Rowe, 2001) creates adaptation case knowledge compiled from the original case-base, and uses it as the knowledge source of a case-based adaptation system. Further work has applied different learning methods to assemble an ensemble of adaptation rule-based adaptation experts learned from these adaptation cases (Wiratunga, Craw & Rowe, 2002).

5 Retention in CBR

In the classic review paper by Aamodt & Plaza (1994), retention is presented as the final step in the CBR cycle, in which the product of the most recent problem-solving episode is incorporated into the system's knowledge. To a great extent this has traditionally translated into a variety of approaches for recording the product of problem solving as a new case that can be added to the case-base. Of course, there are various issues concerning how best to learn a new case and different systems record different types of information in their cases. Most, for example, simply record the target problem specification and the final solution, with the implicit assumption that the outcome was successful. For example, when CBR is integrated with a generative problem-solving system for speed-up learning, the success of the system's solutions may be guaranteed (e.g., Veloso 1992, Veloso & Carbonell 1994). When outcomes are less reliable or when the criteria for success are more complex, case representations must include additional information on the *outcome* of the solution, which may also include fine-grained information on how well the solution addressed many dimensions of the system's goals (e.g., Goel, Kolodner *et al.* 1991). Another question is what to store concerning the solution itself. Many systems store only the solution, but others seek to record a much deeper representation of the problem-solving process that brought about the particular solution. In Veloso & Carbonell's work, for example, *derivational traces* are stored in cases. These rich knowledge structures describe precisely how a given solution was derived, providing a trace of the decision-making processes that led to a particular solution.

In general, the modern view of retention accommodates a much broader perspective of what it means for a CBR system to learn from its problem solving experience, a view that is largely a response to certain critical issues that have arisen during the practical application of CBR

systems in complex problem solving scenarios. In this section we will review this body of work, highlighting many of the critical issues associated with open-ended case learning policies and how these issues have been resolved by novel approaches to case-base optimization. Moreover, we will argue that these issues have served as an important catalyst for research in the area of case-base maintenance and the maintenance of other aspects of a CBR system, which accommodates a broader perspective on case learning and retention.

5.1 *The utility problem in CBR*

In the past the prevailing view of case learning in CBR was based in the assumption that learning would occur as a by-product of every problem-solving episode. However, as CBR systems were developed and deployed for real-world application scenarios, the potential pitfalls of long-term case learning became apparent, especially in relation to the impact of case-base growth on retrieval costs. This so-called *utility problem* for CBR parallels the well-known utility problem identified in explanation-based learning research (e.g., Minton, 1990). The latter refers to the performance degradation experienced by speed-up learners as a result of learning control knowledge. In brief, Minton demonstrated how rules learned with a view to reducing problem solving time, by directing the search more carefully, might ultimately come to degrade overall system performance as the time spent considering the application of a speed-up rule eventually overtook the time needed for from-scratch problem solving. For example, overly specific rules that are seldom applicable, or rules with a high match cost, or rules that offer limited speed-up were all found to contribute to a decline in problem solving efficiency.

At the heart of the utility problem is a natural trade-off between the benefits of speed-up knowledge and the cost of its application and a similar trade-off also exists in CBR systems (e.g., Francis & Ram, 1994; Smyth & Keane, 1995b; Smyth & Cunningham, 1996). Cases correspond to a form of speed-up knowledge, in the sense that retrieval and reuse of similar cases are expected to provide more efficient problem solving than first-principles methods, with additional cases increasing the range of problems that can be solved rapidly. However, this rather naive view of case knowledge is fundamentally flawed by neglecting to consider retrieval costs. In CBR systems the utility problem is caused by the conflict between: (1) the average saving in adaptation effort due to the availability of a particular case, which tends to increase efficiency as the case base grows, and (2) the average retrieval time associated with a given case-base size, which tends to decrease efficiency. Smyth & Cunningham (1996) demonstrate the inevitability of the utility problem in CBR, under reasonable general assumptions about the retrieval and reuse characteristics of a CBR system. They show that as a result of case learning, retrieval efficiency (mean retrieval time) tends to degrade while adaptation efficiency (mean adaptation time) is seen to improve, but at an ever decreasing rate. Initially, as a case base grows each newly learned case can have a significant impact on adaptation as it is more likely to improve overall case-base coverage. However, as the case base grows new cases are more likely to overlap with existing cases and so offer little in the way of new coverage and minimal adaptation savings. As new cases are added retrieval costs become progressively greater but adaptation savings progressively less. Eventually the increase in retrieval time as a result of a new case addition is greater than the adaptation saving offered. At this critical case-base size, overall problem solving efficiency begins to degrade.

5.2 Harmful cases, competence models, and selective retention

Once the relevance of the utility problem to CBR became clear, researchers began to look to the machine learning literature as a source of coping strategies. Several strategies have been proposed to solve the utility problem in machine learning. Markovitch & Scott (1993) propose a unifying framework for the systematic discussion of all of the various strategies for coping with *harmful knowledge* in general, and the utility problem in particular. Their framework is based on different types of filters for eliminating harmful knowledge at various stages in the problem solving cycle. One approach that is especially relevant in CBR is to simply delete harmful cases from the case base so that they cannot actively contribute to ongoing problem solving costs – these deletion approaches correspond to *selective retention filters* in the Markovitch & Scott framework. Surprisingly enough, in many speed-up learners even the apparently naive random deletion of knowledge items (to maintain the knowledge base to some predefined size) works quite well at optimizing efficiency. Even though random deletion removes both useful and redundant items it can equal the success of more sophisticated methods (Markovitch & Scott, 1993). More sophisticated deletion policies have been developed and are guided by some assessment of the utility of individual knowledge items. For example, Minton (1990) uses a utility metric that takes into account the cost of including the item in the set of candidates to consider (match cost) and the expected savings offered by the item (average savings multiplied by its application frequency) to deliver even greater protection against the damaging effects of the utility problem.

Unfortunately it soon became clear that the same type of coping strategies would not translate directly over to case-based reasoners. The problem stems from the fact that many case-based reasoners are not simply using case knowledge as a form of speed-up knowledge. Instead, cases are often a primary source of problem-solving knowledge. Without cases, certain problems cannot be solved and thus the act of deleting cases may irrevocably reduce the competence of the system to solve new problems; CBR systems may not be able to reconstruct deleted cases from an internal domain model. To address this problem, seminal work by Smyth & Keane (1995b) proposed the need for a *competence model* as a way to better evaluate the contributions of individual cases to problem-solving competence. In particular, they developed methods for categorizing cases according to their competence characteristics with a view to guiding the selection of cases for deletion. These categories facilitate the preservation of key cases (called *pivotal cases*) that might otherwise be deleted, in favor of deleting less critical cases whose loss is expected to least harm system competence. Competence-guided case deletion provides for a safe way to eliminate cases from a growing case base, to stave off the harmful effects of the utility problem while at the same time protecting against reductions in competence.

Later work brought the introduction of a more fine-grained model of case competence (Smyth & McKenna, 1998) as a pre-cursor to a variety of related retention models and other forms of case-base editing (e.g., McKenna & Smyth 2000a; 2000b; Smyth & McKenna, 2000a). For example, as an alternative to case deletion, Smyth & McKenna (1999b) use their competence model to develop a competence-guided case addition algorithm. Indeed, related work by Zhu & Yang (1999) describes a case addition algorithm that has the added advantage of providing a guaranteed lower bound on resulting competence. Leake & Wilson (2000) highlight the

importance of considering both competence *and* performance during case-base optimization. They argue the need for more fine-grained performance metrics with which to guide the maintenance of a case base and show how one such metric can help to guide case-base editing in a way that gives due consideration to competing factors such as case-base size, coverage, adaptation performance etc.

Over the past few years there has been a broad range of research addressing these key issues of case deletion, addition and case-based editing in general. Further discussion is beyond the scope of this article but the interested reader is referred to work by Surma & Tyburey (1998), Lei *et al* (1999), Portinale & Torasso (2001), Yang & Zhu (2001), Salamó & Golobardes (2002), Wiratunga *et al.* (2003), and Woon *et al.* (2003).

5.3 Case-base maintenance

As researchers began to recognize that there was more to case retention than simply which cases to learn, and how they should be encoded, the importance of case-base maintenance quickly came into focus (Smyth, 1998; Leake *et al.*, 2001; Wilson & Leake, 2001). Maintenance issues arise when designing and building CBR systems and support tools that monitor system state and effectiveness in order to determine whether, when, and how to update CBR system knowledge to better serve specific performance goals. Understanding the issues that underlie the maintenance problem and using that understanding to develop good practical maintenance strategies is crucial to sustaining and improving the efficiency and solution quality of CBR systems as their case bases grow and as their tasks or environments change over long-term use. And today there is a general recognition of the value of maintenance to the success of practical CBR systems.

To begin to appreciate the issues involved in developing maintenance strategies, as well as to understand maintenance practice and identify opportunities for new research, it is useful to understand the nature of the maintenance process and its relationship to the overall CBR process. Wilson & Leake (2001) characterize case-base maintenance in terms of the components of maintenance policies and the dimensions along which alternative maintenance policies may differ, using this characterization to examine a range of concrete maintenance strategies and proposals. Their framework categorizes case-base maintenance policies in terms of how they gather data relevant to maintenance decisions, how they determine when to trigger maintenance operations, the types of maintenance operations available, and how the selected maintenance operations are executed. For example, data collection might be restricted to gathering information on individual cases (e.g., the number of times a case has been used, or has been used and produced an unsuccessful result) or about the case base as a whole (e.g., its current size, or its growth trends over time). Maintenance policy triggering may be done periodically (e.g., at every case addition), conditionally (e.g., when retrieval time increases to a pre-specified threshold), or on an *ad hoc* basis (e.g., by unpredictable intervention by a human maintainer). The available maintenance operations may target different knowledge containers (e.g., indices, the cases themselves, or adaptation knowledge) and may be applied at different times or to varying portions of the case base. They use this framework to characterize existing strategies according to the framework's dimensions, providing both a snapshot of the current state of the art in case-base maintenance and suggestions of unexplored strategies.

Of course, the success of maintenance depends not only on the maintenance policies themselves, but also on how maintenance is integrated with the overall CBR process. Reinartz, Roth-Berghofer & Iglezakis (2001) propose to extend the classic 4-stage CBR cycle shown in Figure 1 to include two new steps, a *review* step, to monitor the quality of system knowledge, and a *restore* step, which selects and applies maintenance operations. This revised model, shown in Figure 3, emphasizes the important role of maintenance in modern CBR and indeed proposes that the concept of maintenance encompass the retain, review and restore steps.

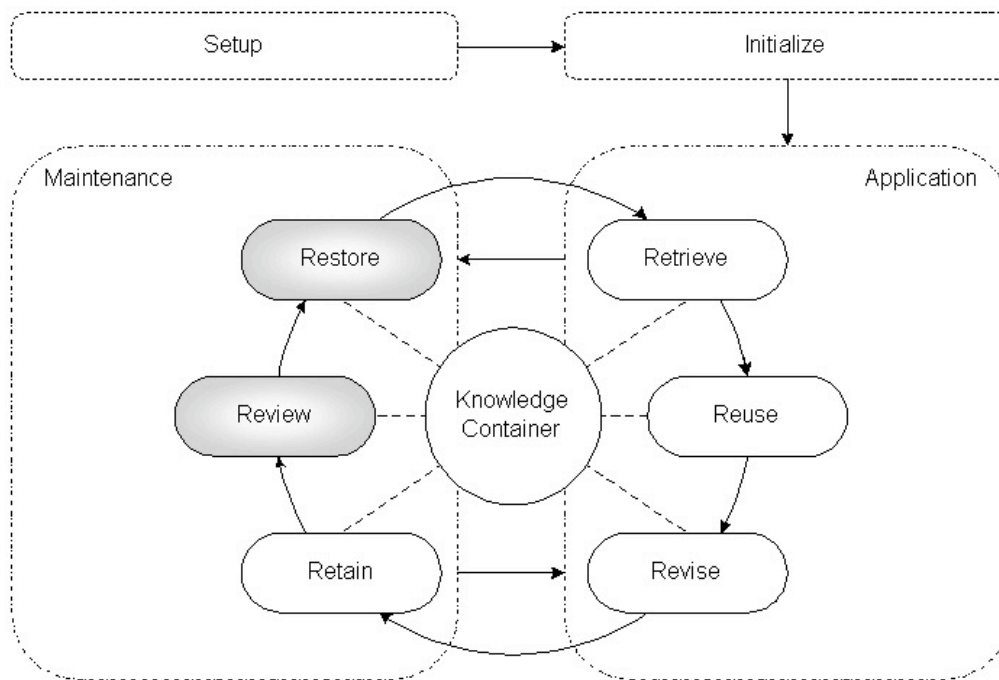


Figure 3 An extension of the classical 4-stage CBR model to emphasize the importance of maintenance in overall system performance, illustrating the setup, initialization, application and maintenance phases of the SIAM methodology for maintaining CBR systems (Iglezakis, Reinartz & Roth-Berghofer, 2004).

A considerable body of maintenance research has obviously developed directly from earlier work on how best to control the addition and deletion of cases in a CBR system (see Section 5.2), but case addition/deletion is just one aspect of maintenance. For example, maintenance policies can be applied to a variety of other knowledge sources beyond the case base. For instance, Hammond (1989) uses explanations of case application failures to determine additional indices to assign to a new case to focus future retrievals. Fox & Leake (1995; 2001) use introspective learning techniques to examine the issue of index refinement triggered by retrieval failures. Munoz-Avila (2001) looks at index revision (and case retention) policies in the context of a derivational replay framework. Index revision is guided by a policy that is based on an analysis of whether the results of retrievals can be extended for new problem scenarios without revising the planning decisions suggested by the retrieved case. Craw, Jarmulak & Rowe (2001) examine the use of a genetic algorithm for refining indexing features and matching weights; see also

Wettschereck & Aha (1995) and Bonzano, Cunningham & Smyth (1997). Maintenance can also involve adaptation. Leake & Wilson (1999) propose adding adaptation rules as a “lazy” strategy for updating the case base as future cases are retrieved, and Shui *et al.* (2001) generate new adaptation rules while compressing the case base as a means to protect against knowledge loss.

In multi-agent scenarios, a CBR system’s own case retention process may be bolstered by drawing on the case bases of cooperating agents, raising questions of when to access those cases and to retain them in the agent’s own case base. This requires strategies for addressing questions such as when external cases may be useful, how to process them to maximize their value to a particular agent, and when multiple case bases should be merged into a single case base (Ontañón & Plaza, 2003; Leake & Sooriamurthi, 2004).

Techniques have been developed for detecting inconsistencies in the case base, either to avoid storing inconsistent cases during initial case retention (McSherry, 1998) or to enable correction of inconsistencies when maintaining the case base as a whole (e.g., Shimazu & Takishima, 1996; Racine & Yang, 1997). More generally, Leake & Wilson (1999) look at the use of CBR in changing environments where key challenges exist in relation to the predictability of problem-solving regularity and distribution. They argue that to avoid inconsistent problem-solving performance a CBR system must be able to examine how well these key regularity assumptions hold and take corrective maintenance action when they do not. The study of case retention is therefore inextricably tied to many related issues for managing the multiple forms of knowledge within CBR systems and adapting CBR systems to the needs of the environments in which they function.

Maintenance strategies can also be used to assist the case author during the early stages of case acquisition. For example, Ferrario & Smyth (2001) describe a distributed approach to case authoring in which a community of authors contribute to the validation of new case knowledge. McSherry (2001c) also focuses on the case acquisition task, and presents a system that performs background reasoning on behalf of the case author while new cases are being added, in order to help the user determine the best cases to add in light of their competence contributions. The system uses its evaluations of the contributions of potential cases to suggest cases to add to the case library. McKenna & Smyth (2001) propose an approach to providing authoring support that attempts to identify *competence holes* within an evolving case-base. They demonstrate how their model of competence (Smyth & McKenna, 1998; 2001) can be used to prioritise gaps in case knowledge and, like McSherry (2001c), propose a technique for automatically suggesting the type of cases that an author might want to consider to fill these gaps with a view to maximizing the potential coverage and contributions that are available. To provide a systematic framework for organizations needing to capture and maintain case-based knowledge, work by Nick, Althoff & Tautz (2001) develops systematic practical strategies for guiding the maintenance of corporate experience repositories.

In this section we have attempted to summarize research in the area of retention and maintenance. Due to space limitations, it has only been possible to scratch the surface of this dynamic and rich area of research. Retention and case-base editing and, more generally, case-base maintenance, continues to be a rich source of research ideas, and even recent developments could not be discussed in the detail they deserve in this article. The interested reader is referred to Wilson & Leake (2001) for a thorough examination of the dimensions of maintenance strategies

and survey of additional maintenance research in terms of those dimensions. In addition, a recent collection of maintenance articles addressing numerous facets of maintenance is available in Leake, Smyth, Yang & Wilson (2001).

6 Conclusions

Our aim in this paper has been to provide a concise overview of the cognitive science foundations of CBR and of the four main tasks involved in the CBR cycle, namely retrieval, reuse, revision, and retention. Rather than presenting a comprehensive survey, we have focused on a representative selection of work from the CBR literature over the past couple of decades. We have tried to strike a balance between research that can be seen as laying the foundations of CBR and more recent contributions. The fact that a considerable portion of the discussed papers has been published in the last few years is evidence of a significant amount of ongoing research activity. It should be clear from our discussion that much of the recent research has been motivated by an increased awareness of the limitations of traditional approaches to retrieval, reuse, and retention. This is a trend that seems likely to continue with the emergence of new and more demanding applications of CBR, and we look forward to the challenges and opportunities that lie ahead.

References

- Aamodt, A. 1994 Explanation-driven case-based reasoning. In Wess, S., Althoff, K.-D. & Richter, M. (eds.), *Topics in Case-Based Reasoning*. Springer-Verlag, pp. 274-288.
- Aamodt, A. 2004 Knowledge-intensive case-based reasoning in Creek. In *Proceedings of the Seventh European Conference on Case-Based Reasoning*. Springer-Verlag, pp. 1-15.
- Aamodt, A. & Plaza, E. 1994 Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications* 7(1), 39-59.
- Aha, D. W., Breslow & Muñoz-Avila, H. 2001 Conversational case-based reasoning. *Applied Intelligence* 14(1), 9-32.
- Aleven, V., 2003 Using background knowledge in case-based legal reasoning: A computational model and an intelligent learning environment. *Artificial Intelligence* 150, 183-237.
- Ashley, K. D. 1989 Defining salience in case-based arguments. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, pp. 537-542.
- Ashley, K. D. 1991 Reasoning with cases and hypotheticals in Hypo. *International Journal of Man-Machine Studies* 34, 753-796.
- Ashley, K. D. & Aleven, V. 1992 Generating dialectical examples automatically. In *Proceedings of the Tenth National Conference on Artificial Intelligence*. AAAI Press/The MIT Press, pp. 654 -660
- Ashley, K.D. & Aleven, V. 1997 Reasoning symbolically about partially matched cases. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, pp. 335-341.
- Arcos, J. L. 1997 *The NOOS Representation Language*. PhD Thesis, Artificial Intelligence Research Institute and Technical University of Catalonia. IIIA-CSIC Monographies Series.
- Arcos, J. L. & López de Mántaras, R (1997), Perspectives: A declarative bias mechanism for case retrieval. In *Proceedings of the Second International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 279-290.
- Arcos, J. L. & López de Mántaras, R. 2001 An interactive case-based reasoning approach for generating expressive music. *Applied Intelligence* 14(1), 115-129.
- Bergmann, R. 2002 *Experience Management*. Springer-Verlag.
- Bergmann R., Richter, M. M., Schmitt, S., Stahl, A. & Vollrath, I. 2001 Utility-oriented matching: A new research direction for case-based reasoning. In *Proceedings of the Ninth German Workshop on Case-Based Reasoning*.
- Bergmann R. & Stahl, A. 1998 Similarity measures for object-oriented case representations. In *Proceedings*

- of the Fourth European Workshop on Case-Based Reasoning. Springer-Verlag, pp. 25-36.
- Bergmann, R., Vollrath, I. & Wahlmann, T. 1999 Generalized cases and their application to electronic design. In *Proceedings of the Seventh German Workshop on Case-Based Reasoning*.
- Bichindaritz, I. 1995 Case-based reasoning adaptive to several cognitive tasks. In *Proceedings of the First International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 391-400.
- Bogaerts, S. & Leake, D. 2004 Facilitating CBR for incompletely-described cases: distance metrics for partial problem descriptions. In *Proceedings of the Seventh European Conference on Case-Based Reasoning*. Springer-Verlag, pp. 62-76.
- Bonzano, A., Cunningham, P. & Smyth, B. 1997. Using introspective learning to improve retrieval in CBR: A case study in air traffic control. *Proceedings of the Second International Conference on Case-Based Reasoning*. Springer Verlag, pp. 291-302.
- Börner, K. 1993 Structural similarity as guidance in case-based design. In *Proceedings of the First European Workshop on Case-Based Reasoning*. Springer-Verlag, pp. 197-208.
- Bower, G., Black, J. & Turner, T. 1979 Scripts in memory for text. *Cognitive Psychology* 11, pp. 177-220.
- Bridge, D. & Ferguson, A. 2002a An expressive query language for product recommender systems. *Artificial Intelligence Review* 18(3-4), 269-307.
- Bridge, D. & Ferguson, A. 2002b Diverse product recommendations using an expressive language for case retrieval. In *Proceedings of the Sixth European Conference on Case-Based Reasoning*. Springer-Verlag, pp. 43-57.
- Bunke, H. & Messmer, B. T. 1993 Structural similarity as guidance in case-based design. In *Proceedings of the First European Workshop on Case-Based Reasoning*. Springer-Verlag, pp. 106-118.
- Branting, L.K. 1991 Building Explanations from Rules and Structured Cases, *International Journal of Man-Machine Studies* 34, 797-837.
- Brown, M.G. 1994 An underlying memory model to support case retrieval. In Wess, S., Althoff, K.-D. & Richter, M. (eds.), *Topics in Case-Based Reasoning*. Springer-Verlag, pp. 132-143.
- Carbonell, J. G. 1986 Derivational analogy: A theory of reconstructive problem solving and expertise acquisition. In Michalski, R.S., Carbonell, J.G. & Mitchell, T.M. (eds.), *Machine Learning: An Artificial Intelligence Approach Volume II*. Los Altos, CA : Morgan Kaufmann, pp. 371-392.
- Champin, P. A. & Solnon, C. 2003 Measuring the similarity of labeled graphs. In *Proceedings of the Fifth International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 80-95.
- Cohen, Paul R. (1985). *Heuristic Reasoning about Uncertainty: An Artificial Intelligence Approach*. Pitman: Boston.
- Craw, S., Jarmulak, J., & Rowe, R. 2001a Learning and applying case-based adaptation knowledge. In *Proceedings of the Fourth International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 131-145.
- Craw, S., Jarmulak, J. & Rowe, R. 2001b Maintaining retrieval knowledge in a case-based reasoning system. Special Issue on Maintaining Case-Based Reasoning Systems, *Computational Intelligence* 17(2), 346-363.
- Cunningham, P., Doyle, D. & Loughrey, J. 2003 An evaluation of the usefulness of case-based explanation. In *Proceedings of the Fifth International Conference on Case-Based Reasoning*. Springer, pp. 122-130.
- Domeshek, E. 1992 *Do the Right Things: A Component Theory for Indexing Stories as Social Advice*. PhD Dissertation, Institute for the Learning Sciences, Northwestern University, Evanston, IL.
- Doyle, D., Cunningham, P., Bridge, D. & Rahman, Y. 2004 Explanation oriented retrieval. In *Proceedings of the Seventh European Conference on Case-Based Reasoning*. Springer-Verlag, pp. 157-168.
- Emde, W. & Wettschereck, D. 1996 Relational instance-based learning. In *Proceedings of the Thirteenth International Conference on Machine Learning*. Morgan Kaufmann, pp. 122-130.
- Falkenhainer, B., Forbus, K. & Gentner, D. 1986 The structure-mapping engine. In *Proceedings of AAAI-86*, pp. 272-277.
- Faltings, B. 1997a Case reuse by model-based interpretation. In Maher, M. L. & Pu, P. (eds.), *Issues and Applications of Case-Based Reasoning in Design*, Lawrence-Erlbaum, pp. 39-60.
- Faltings, B. 1997b Probabilistic indexing for Case-Based prediction, In *Proceedings of the Second International Conference on Case-Based Reasoning*, Springer Verlag LNAI 1266, pp. 611-622.
- Faltings, B. & Sun, K. 1996 FAMING: Supporting innovative mechanism shape design. *Computer-Aided Design* 28(3), 207-216.

- Ferrario, M-A. & Smyth, B. 2001 Distributing case-base maintenance: The collaborative maintenance approach. Special Issue on Maintaining Case-Based Reasoning Systems, *Computational Intelligence* 17(2), 315-330.
- Forbus, K. 2001 Exploring analogy in the large. In Gentner, D., Holyoak, K. & Kokinov, B. (eds.), *The Analogical Mind: Perspectives from Cognitive Science*. Cambridge, MA: MIT Press, pp. 23-58.
- Forbus, K., Ferguson, R. & Gentner, D. 1994 Incremental structure-mapping. In *Proceedings of the Cognitive Science Society*, August, pp. 225-230.
- Forbus, K., Gentner, D. & Law, K. 1994 MAC/FAC: A model of similarity-based retrieval. *Cognitive Science* 19(2), 141-205.
- Forbus, K., Usher, J. & Chapman, V. 2003 Qualitative spatial reasoning about sketch maps. In *Proceedings of the Fifteenth Annual Conference on Innovative Applications of Artificial Intelligence*. Morgan Kaufmann, pp 85-92.
- Fox, S. & Leake, D. 1995 Using introspective reasoning to refine indexing. *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, pp. 391-399.
- Fox, S., & Leake, D. 2001 Introspective reasoning for index refinement in case-based reasoning. *Journal of Experimental and Theoretical Artificial Intelligence* 13(1), 63-88.
- Francis, A. G. & Ram, A. 1994 A Comparative utility analysis of case-based reasoning and control-rule learning systems. In *Proceedings of the European Conference on Machine Learning (ECML-95)*, pp. 138-150.
- Fuchs, B. & Mille, A. 1999 A knowledge-level task model of adaptation in case-based reasoning. In *Proceedings of the Third International Conference on Case Based Reasoning*. Springer-Verlag, pp. 118-131.
- Gabel, T & Stahl, A. 2004 Exploiting background knowledge when learning similarity measures. In *Proceedings of the Seventh European Conference on Case-Based Reasoning*. Springer-Verlag, pp. 169-183.
- Gentner, D. 1983 Structure-mapping: A theoretical framework for analogy. *Cognitive Science* 7, pp. 155-170.
- Gentner, D. & Forbus, K. 1991 MAC/FAC: A model of similarity-based retrieval. In *Proceedings of the Thirteenth Cognitive Science Conference*. Hillsdale, NJ: Erlbaum, pp. 504-509.
- Gentner, D., Holyoak, K. & Kokinov, B. (eds.) 2001 *The Analogical Mind: Perspectives from Cognitive Science*. Cambridge, MA: MIT Press.
- Gentner, D., Rattermann, M. J. & Forbus, K. 1993 The roles of similarity in transfer: Separating retrieval from inferential soundness. *Cognitive Psychology* 25, pp. 524-575.
- Gick, M. L. & Holyoak, K.J. 1980 Analogical problem solving. *Cognitive Psychology* 12, pp. 306-355.
- Goel, A. & Chandrasekaran, B. 1989 Use of device models in adaptation of design cases. In *Proceedings of the Fifth National Conference on Artificial Intelligence*. Morgan Kaufmann, pp. 100-109.
- Goel, A., Kolodner, J., Pearce, M., Billington, R. & Zimring, C. 1991. Towards a case-based tool for aiding conceptual design problem solving. In Bareiss, R. (ed.), *Proceedings of the Workshop on Case-Based Reasoning, (DARPA)*. San Mateo, CA: Morgan Kaufmann, pp. 109-120.
- Gómez de Silva Garza, A. & Maher, M. L. 2000 A process model for evolutionary design case adaptation. In *Proceedings of the Artificial Intelligence in Design Conference*. Kluwer Academic Publishers, pp. 393-412.
- Hammond, K. J. 1986 CHEF: A model of case-based planning. In *Proceedings of the Fifth National Conference on Artificial Intelligence*. AAAI Press, pp. 267-271.
- Hammond, K. J. 1990 Explaining and repairing plans that fail. *Artificial Intelligence* 45, pp. 173-228.
- Hanney, K. & Keane, M. T. 1997 The adaptation knowledge bottleneck: How to ease it by learning from cases. In *Proceedings of the Second International Conference on Case Based Reasoning*, Springer-Verlag, pp. 359-370.
- Iglezakis, I. Reinartz, T. & Roth-Berghofer, T. 2004 Maintenance memories: beyond concepts and techniques for case base maintenance. In *Proceedings of the Seventh European Conference on Case-Based Reasoning*, Springer-Verlag, pp. 227-241.
- Jantke, K.P. 1994 Non-standard concepts in similarity in case-based reasoning. In *Proceedings der Jahrestagung der Gesellschaft für Klassifikation*. Springer-Verlag, pp. 29-44.
- Jarmulak, J., Craw, S. & Rowe, R. 2001 Using case-base data to learn adaptation knowledge for design. In

- Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, pp. 1011–1016.
- Kass, A. 1989 Strategies for adapting explanations. In *Proceedings of the Second DARPA Workshop on Case-Based Reasoning*. Morgan Kaufmann, pp. 119–123.
- Kass, A., Leake, D. & Owens, C. 1986 SWALE: A program that explains. In Schank, R. (ed.) *Explanation Patterns: Understanding Mechanically and Creatively*. Lawrence Erlbaum, pp. 232-254.
- Keane, M. 1988 *Analogical Problem Solving*. Chichester: Ellis Horwood (New York: Wiley).
- Keane, M. & Brayshaw, M. 1988 The incremental analogy machine: a computational model of analogy. *Proceedings of the Third European Working Session on Learning*. Pitman, pp. 53-62.
- Keane, M T, Ledgeway, T, & Duff, S. 1994. Constraints on analogical mapping: A comparison of three models. *Cognitive Science* 18, 287-334.
- Keil, F. & Wilson, R. A. 2000 Explaining explanation. In Keil, F. & Wilson, R. A. (eds.), *Explanation and Cognition*. Cambridge, MA: MIT Press, pp. 1-18.
- Kokinov, B. & French, R. 2002 Computational models of analogy-making. In Nadel, L. (ed.) *Encyclopedia of Cognitive Science*. Macmillan, pp. 113-118.
- Kolodner, J. L. 1991 Improving human decision making through case-based decision aiding, *AI Magazine* 12(2), pp. 52–68.
- Kolodner, J. L. 1993 *Case-Based Reasoning*. San Mateo, CA: Morgan Kaufmann.
- Kolodner, J. L. 1994 From natural language understanding to case-based reasoning and beyond: A perspective on the cognitive model that ties it all together. In Schank, R. C. & Langer, E. (eds.), *Beliefs, Reasoning, and Decision Making: Psycho-Logic in Honor of Bob Abelson*. Hillsdale, NJ: Erlbaum.
- Kolodner, J. L. 1996 Making the implicit explicit: Clarifying the principles of case-based reasoning. In Leake, D. B. (ed.) *Case-Based Reasoning: Experiences, Lessons & Future Directions*. MIT Press, pp. 349-370.
- Kolodner, J. L., Camp, P. J., Crismond, D., Fasse, B., Gray, J., Holbrook, J. & Puntembakar, S. 2003 Problem-based learning meets case-based reasoning in the middle-school science classroom: Putting learning-by-design into practice. *Journal of the Learning Sciences* 12(4), pp. 495-547.
- Koton. P. 1988 Reasoning about evidence in causal explanations. In *Proceedings of the Seventh National Conference on Artificial Intelligence*. AAAI Press, pp. 256-261.
- Larkey, L B. & Love, B. C. 2003. CAB: Connectionist analogy builder. *Cognitive Science* 27, 781-794.
- Leake, D.B. 1991 Goal-Based Explanation Evaluation. *Cognitive Science* 15(4), 509-545.
- Leake, D. B. 1992 *Evaluating Explanations: A Content Theory*. Hillsdale, NJ: Erlbaum.
- Leake, D. B. 1996 CBR in context: The present and future. In Leake, D. B. (ed.) *Case-Based Reasoning: Experiences, Lessons & Future Directions*. MIT Press, pp. 3-30.
- Leake, D. B. 1998 Cognition as case-based reasoning. In Bechtel, W. & Graham, G. (eds.), *A Companion to Cognitive Science*. Oxford: Blackwell, pp. 465-476.
- Leake, D. B., Kinley, A. & Wilson, D. 1995 Learning to improve case adaptation by introspective reasoning and CBR. In *Proceedings of the First International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 229-240.
- Leake, D. B., Kinley, A. & Wilson, D. 1996 Acquiring case adaptation knowledge: A hybrid approach. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*. AAAI Press, pp. 684–689.
- Leake, D. B., Kinley, A. & Wilson, D. 1997 Learning to integrate multiple knowledge sources for case-based reasoning. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, pp. 674-679.
- Leake, D., Smyth, B., Wilson, D. & Yang, Q. 2001 Introduction to the special issue on maintaining case-based reasoning systems. *Computational Intelligence* 17(2), 193-195.
- Leake, D. & Sooriamurthi, R. 2004 Case dispatching versus case-base merging: When MCBR matters. *International Journal of Artificial Intelligence Tools* 13(1), 237-254.
- Leake, D. & Wilson, D. 1999 When experience is wrong: Examining CBR for changing tasks and environments. *Proceedings of the Third International Conference on Case-Based Reasoning*. Springer Verlag, pp. 218-232.
- Leake, D. & Wilson, D. 2000 Remembering why to remember: Performance-guided case-base maintenance. *Proceedings of the Fifth European Workshop on Case-Based Reasoning*. Springer Verlag, pp. 161-172.
- Lei, C., Babka, O. & Garanito, L.A.G. 1999 Case library reduction applied to pile foundations. *Proceedings*

- of the *Third International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 233-247.
- Lenz, M. 1996 Applying case retrieval nets to diagnostic tasks in technical domains. In Smith, I. & Faltings, B. (eds.), *Advances in Case-Based Reasoning*. Springer-Verlag, pp. 219-233.
- Larkey, L. B. & Love, B. C. 2003 CAB: Connectionist analogy builder. *Cognitive Science* 27, pp. 781-794.
- López de Mántaras, R. & Arcos, J. L. 2002 AI and music: From composition to expressive performance. *AI Magazine* 23(3), 43-57.
- Markovitch, S. & Scott, P. D. 1993 Information filtering: selection mechanisms in learning systems. *Machine Learning* 10, pp. 113-151.
- McGinty, L. & Smyth, B. 2003) On the role of diversity in conversational recommender systems. *Proceedings of the Fifth International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 276-290.
- McKenna, E., & Smyth, B.: Competence-guided case discovery. In *Proceedings of the Twenty-First BCS SGES International Conference on Knowledge Based Systems and Applied Artificial Intelligence*, pp. 97-108.
- McKenna, E. & Smyth, B. 2000a Competence-guided editing methods for lazy learning. In *Proceedings of the Fourteenth European Conference on Artificial Intelligence*, IOS Press, pp. 60-64
- McKenna, E. & Smyth, B. 2000b competence-guided case-base editing techniques. In *Proceedings of the Fifth European Workshop on Case-Based Reasoning*. Springer-Verlag, pp. 186-197.
- McSherry, D. 1998 An adaptation heuristic for case-based estimation. In *Proceedings of the Fourth European Workshop on Case-Based Reasoning*. Springer-Verlag, pp. 184-195.
- McSherry, D. 2001a Minimizing dialog length in interactive case-based reasoning. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, pp. 993-998.
- McSherry, D. 2001b Precision and recall in interactive case-based reasoning. In *Proceedings of the Fourth International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 392-406.
- McSherry, D. 2001c Intelligent case-authoring support in CaseMaker-2. *Computational Intelligence* 17(2), 331-345.
- McSherry, D. 2001d Interactive case-based reasoning in sequential diagnosis. *Applied Intelligence* 14(1), 65-76.
- McSherry, D. 2002 Diversity-conscious retrieval. In *Proceedings of the Sixth European Conference on Case-Based Reasoning*. Springer-Verlag, pp. 219-233.
- McSherry, D. 2003a Explanation of retrieval mismatches in recommender system dialogues. In *Proceedings of the ICCBR 2003 Workshops*. Norwegian University of Science and Technology, Department of Computer and Information Science, pp. 191-199.
- McSherry, D. 2003b Increasing dialogue efficiency in case-based reasoning without loss of solution quality. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, pp. 121-126.
- McSherry, D. 2003c Similarity and compromise. In *Proceedings of the Fifth International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 291-305.
- McSherry, D. 2004a Explaining the pros and cons of conclusions in CBR. In *Proceedings of the Seventh European Conference on Case-Based Reasoning*. Springer-Verlag, pp. 317-330.
- McSherry, D. 2004b Explanation in recommender systems In *Proceedings of the ECCBR 2004 Workshops* (Technical Report 142-04). Universidad Complutense de Madrid, Departamento de Sistemas Informáticos y Programación, pp. 125-134.
- Minton, S. 1990 Qualitative results concerning the utility of explanation-based learning. *Artificial Intelligence* 42, 363-391.
- Mougouie, B. & Bergmann, R. 2002 Similarity assessment for generalized cases by optimization methods. In *Proceedings of the Sixth European Conference on Case-Based Reasoning*. Springer-Verlag, pp. 249-263.
- Muñoz-Avila, H. & Hüllen, J. 1996 Feature weighting by explaining case-based planning episodes. In *Proceedings of the Third European Workshop on Case-Based Reasoning*. Springer-Verlag, pp. 280-294.
- Munoz-Avila, H. 2001 Case-base maintenance by integrating case index revision and case retention policies in a derivational replay framework. Special Issue on Maintaining Case-Based Reasoning Systems, *Computational Intelligence* 17(2), 280-294.
- Nick, M., Althoff, K.-D. & Tautz, C. 2001 Systematic maintenance of corporate experience repositories.

- Computational Intelligence* 17(2), 364-386.
- Ontañón, S. & Plaza, E. 2003 Collaborative case retention strategies for CBR agents. In *Proceedings of the Fifth International Conference on Case-Based Reasoning*, Springer-Verlag, pp. 392-406.
- Osborne, H. R. & Bridge, D. G. 1996 A case base similarity framework. In *Proceedings of the Third European Workshop on Case-Based Reasoning*. Springer-Verlag, pp. 309-323.
- Osborne, H. R. & Bridge, D. G. 1997 Similarity metrics: A formal unification of cardinal and non-cardinal similarity measures. In *Proceedings of the Second International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 235-244.
- Ozturk, P. & Aamodt, A. 1998 A context model for knowledge-intensive case-based reasoning. *International Journal of Human Computer Studies* 48, 331-355.
- Patel, V. L. & Groen, G. J. 1986. Knowledge-based solution strategies in medical reasoning. *Cognitive Science* 10, 91-116,
- Plaza, E. 1995 Cases as terms: A feature term approach to the structured representation of cases. In *Proceedings of the First International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 265-276.
- Porter B., Bareiss R. & Holte R. 1990 Concept learning and heuristic classification in weak theory domains. *Artificial Intelligence* 45(1-2), 229-263.
- Portinale, L. & Torasso, P. 2001 Case-base maintenance in a multi-modal reasoning system. *Computational Intelligence* 17(2), 263-279.
- Purvis, L. & Pu, P. 1995 Adaptation using constraint satisfaction techniques. In *Proceedings of the First International Conference on Case Based Reasoning*. Springer-Verlag, pp. 289-300.
- Racine, K. & Yang, Q. 1997 Maintaining unstructured case bases. *Proceedings of the Second International Conference on Case-Based Reasoning*, Springer-Verlag, pp. 553-564.
- Ramscar, M. & Yarlett, D. 2003 Semantic grounding in models of analogy: An environmental approach. *Cognitive Science* 27, pp. 41-71.
- Reinartz, T., Iglezakis, I. & Roth-Berghofer, T. 2001. Review and restore for case-based maintenance. *Computational Intelligence* 17(2), 214-234.
- Richter, M. M. 1992 Classification and learning of similarity measures. In *Proceedings der Jahrestagung der Gesellschaft für Klassifikation*. Springer-Verlag, pp. 1-8.
- Rissland, E., Valcarce, E. & Ashley, K. 1985 Explaining and arguing with examples. In *Proceedings, Fourth National Conference on Artificial Intelligence*. Morgan Kaufmann, pp. 288-294.
- Rissland, E.L. & Skalak, D.B. 1991 CABARET: Statutory interpretation in a hybrid architecture. *International Journal of Man-Machine Studies* 34(6), 839-887.
- Salamó, M. & Golobardes, E. 2002 Deleting and building sort out techniques for case-base maintenance. In *Proceedings of the Sixth European Conference on Case-Based Reasoning*. Springer-Verlag, pp.365-379.
- Schaaf, J. W. 1996 Fish & shrink: A next step towards efficient case retrieval in large-scale case-bases. In Smith, I. & Faltings, B. (eds.), *Advances in Case-Based Reasoning*. Berlin Heidelberg New York: Springer-Verlag, pp. 362-376.
- Schank, R.C. 1982 *Dynamic Memory: A Theory of Reminding and Learning in Computers and People*. Cambridge, MA: Cambridge University Press.
- Schank, R.C. 1986 *Explanation Patterns: Understanding Mechanically and Creatively*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Schank, R.C. & Abelson, R. P. 1977 *Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Schank, R.C., Fano, A., Bell, B. & Jona, M. 1993 The design of goal-based scenarios. *Journal of the Learning Sciences* 3(4), pp. 305-345.
- Schank, R.C., Kass, A. & Riesbeck, C. K. 1994 *Inside Case-Based Explanation*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Schank, R.C. & Leake, D. 1989 Creativity and learning in a case-based explainer. *Artificial Intelligence* 40(1-3), pp. 353-385.
- Schank, R.C., Osgood, R., Brand, M., Burke, R., Domeshek, E., Edelson, D., Ferguson, W., Freed, M., Jona, M., Krulwich, B., Ohmayo, E. & Pryor, L. 1992 *A Content Theory of Memory Indexing*. Technical Report 1, Institute for the Learning Sciences, Northwestern University, Evanston, IL.
- Shih, J. 2001 Sequential instance-based learning for planning in the context of an imperfect information

- game. In *Proceedings of the Fourth International Conference on Case-Based Reasoning*. Springer Verlag, pp. 483-501.
- Shimazu, H. & Takashima, Y. 1996 Detecting discontinuities in case-bases. In *Proceedings of AAAI/IAAI*, Vol. 1, pp. 690-695.
- Shiu, S.C.K., Sun C.H., Wang Z.Z., & Yeung D.S. 2001 Transferring case knowledge to adaptation knowledge: An approach for case-base maintenance. *Computational Intelligence* 17(2), 214-234.
- Schmidt R, Montani S, Bellazzi R, Portinale L & Gierl, L. 2001 Cased-based reasoning for medical knowledge-based systems. *International Journal of Medical Informatics* 64(2-3), 355-67.
- Simoudis, E. & Miller, J. 1990 Validated retrieval in case-based reasoning. In *Proceedings of the Eighth National Conference on Artificial Intelligence*. AAAI Press, pp. 310-315.
- Smyth, B. 1998 Case-base maintenance. In *Proceedings of the Eleventh International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems*. pp. 507-516.
- Smyth, B & Cunningham, P. 1996. The utility problem analysed: A case-based reasoning perspective. In *Proceedings of the Third European Workshop on Case-Based Reasoning*. Springer-Verlag, pp. 392-399.
- Smyth, B. & Keane, M. 1994. Retrieving adaptable cases. In Wess, S., Althoff, K.-D. & Richter, M. (eds.), *Topics in Case-Based Reasoning*. Springer-Verlag, pp. 209-220.
- Smyth, B. & Keane, M. T. 1995a Experiments on adaptation-guided retrieval in case-based design. In *Proceedings of the First International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 313-324.
- Smyth, B. & Keane, M. T. 1995b Remembering to forget: A competence-preserving case deletion policy for case-based reasoning systems. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, pp. 377-383.
- Smyth, B. & Keane, M. T. 1996 Using adaptation knowledge to retrieve reusable designs. *Knowledge-Based Systems* 9(2), 127-135.
- Smyth, B. & Keane, M. T. 1998 Adaptation-guided retrieval: Questioning the similarity assumption in reasoning. *Artificial Intelligence* 102(2), 249-293.
- Smyth, B., Keane, M. T. & Cunningham, P. 2001 Hierarchical case-based reasoning: Integrating case-based and decompositional problem-solving techniques for plant-control software design. *IEEE Transactions on Knowledge and Data Engineering* 13, 793-812.
- Smyth, B. & McClave, P. 2001 Similarity vs. diversity. In *Proceedings of the Fourth International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 347-361.
- Smyth, B. & McKenna, E. 1998 A portrait of case competence: Modelling the competence of case-based reasoning systems. In *Proceedings of the Fourth European Workshop on Case-Based Reasoning*. Springer-Verlag, pp. 208-220.
- Smyth, B. & McKenna, E. 1999a Footprint based retrieval. In *Proceedings of the Third International Conference on Case-based Reasoning*. Springer-Verlag, pp 343-357.
- Smyth, B. & McKenna, E. 1999b Building compact competent case-bases. In *Proceedings of the Third International Conference on Case-based Reasoning*. Springer-Verlag, pp. 329-342.
- Smyth, B. & McKenna, E. 2000 An efficient and effective procedure for updating a competence model for case-based reasoners. In *Proceedings of the Eleventh European Conference on Machine Learning*. Springer-Verlag, pp. 357-368. .
- Smyth, B. & McKenna, E. 2001a Competence models and the maintenance problem. Special Issue on Maintaining Case-Based Reasoning Systems. *Computational Intelligence* 17(2), 235-249.
- Smyth, B. & McKenna, E. 2001b Competence guided incremental footprint-based retrieval. *Knowledge-Based Systems* 14(3-4), 155-161.
- Sørmø, F. & Cassens, J. 2004 Explanation goals in case-based reasoning. In *Proceedings of the ECCBR 2004 Workshops* (Technical Report 142-04). Universidad Complutense de Madrid, Departamento de Sistemas Informáticos y Programación, pp. 165-174.
- Sørmø, F., Cassens, J. & Aamodt, A. 2005/6 Explanation in case-based reasoning; perspectives and goal. To appear in *Artificial Intelligence Review* 24.
- Stahl, A. & Gabel, T. 2003 Using evolution programs to learn local similarity measures. In *Proceedings of the Fifth International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 537-551.
- Stanfill, C. & Waltz, D. 1986 Toward memory-based reasoning. *Communications of the ACM* 29(12), pp. 1213-1228.

- Surma, J. & Tyburcy, J. 1998 A study of competence-preserving case-replacing strategies in case-based reasoning. *Proceedings of the Fourth European Workshop on Case-Based Reasoning*. Springer-Verlag, pp. 233-238.
- Tambe, N., Newell, A. & Rosenbloom, P. S. 1990 The problem of expensive chunks and its solution by restricting expressiveness. *Machine Learning*, 5, pp. 299-349.
- Tartakovski, A., Schaaf, M., Maximini, R. & Bergmann, R. 2004 MINLP based retrieval of generalized cases. In *Proceedings of the Seventh European Conference on Case-Based Reasoning*. Springer-Verlag, pp. 404-418.
- Tversky, A. 1977 Features of similarity. *Psychological Review* 84(4), 327-352.
- Veloso, M. 1992 *Learning by Analogical Reasoning in General Problem Solving*. PhD Thesis, Carnegie Mellon University, USA.
- Veloso, M. & Carbonell, J. 1993 Derivational analogy in PRODIGY: Automating case acquisition, storage, and utilization. *Machine Learning* 10(3), pp. 249-278.
- Veloso, M. & Carbonell, J. G. 1994 Case-based reasoning in Prodigy. In Michalski, R. & Tecuci, G. (eds.), *Machine Learning: A Multistrategy Approach Volume IV*, Morgan Kaufmann, pp. 523-548.
- Veloso, M., Carbonell, J., Alicia Pérez, M., Borrajo, D., Fink, E. & Blythe, J. 1995 Integrating planning and learning: The PRODIGY architecture. *Journal of Experimental and Theoretical Artificial Intelligence*, 7(1), pp. 81-120.
- Watson, I. 1997 *Applying case-based reasoning: Techniques for enterprise systems*. San Francisco, CA: Morgan Kaufmann.
- Wess, S., Althoff, K-D. & Derwand, G. 1993 Using K-D trees to improve the retrieval step in case-based reasoning. In *Proceedings of the First European Workshop on Case-Based Reasoning*. Springer-Verlag, pp. 167-181.
- Wettschereck, D. & Aha, D. 1995 Weighting features. In *Proceedings of the First International Conference on Case-Based Reasoning*. Springer-Verlag, pp. 347-358.
- Wilke, W. & Bergmann, R. 1996 Considering decision cost during learning of feature weights. In *Proceedings of the Third European Workshop on Case-Based Reasoning*. Springer-Verlag, pp. 460-472.
- Wilke, W., Vollrath, I., Althoff, K.-D. & Bergmann, R. 1997 A framework for learning adaptation knowledge based on knowledge light approaches. In *Proceedings of the Fifth German Workshop on Case-Based Reasoning*, pp. 235-242.
- Wilson, D. & Leake, D. 2001 Maintaining case-based reasoners: Dimensions and directions. *Computational Intelligence* 17(2), pp. 196-213.
- Wiratunga, N., Craw, S. & Massie, S. 2003 Index driven selective sampling for CBR. *Proceedings of the Fifth International Conference on Case-Based Reasoning*, pp. 637-651.
- Wiratunga, N., Craw, S. & Rowe, R. 2002 Learning to adapt for case-based design. In *Proceedings of the Sixth European Conference on Case-Based Reasoning*. Springer, pp. 423-437.
- Wolverton, M. & Hayes-Roth, B. 1994 Retrieving semantically distant analogies with knowledge-directed spreading activation. In: *Proceedings of the Twelfth National Conference on Artificial Intelligence*. AAAI Press, pp. 56-61.
- Woon, F. L., Knight, B. & Petridis, P. 2003 Case base reduction using solution-space metrics. *Proceedings of the Fifth International Conference on Case-Based Reasoning*, Springer-Verlag, pp. 652-664.
- Yang, Q. & Zhu, J. 2001 A case-addition policy for case-base maintenance. *Computational Intelligence* 17(2), pp. 250-262.
- Zhu, J. & Yang, Q. 1999 Remembering to add: Competence-preserving case-addition policies for case base maintenance. *International Joint Conference on Artificial Intelligence*, Morgan Kaufmann, pp. 234-241.