Learning More from Experience in Case-Based Reasoning

David McSherry and Christopher Stretch

School of Computing and Information Engineering,
University of Ulster, Coleraine BT52 1SA, Northern Ireland
{dmg.mcsherry,ct.stretch}@ulster.ac.uk

Abstract. Recent concerns about the effects of feedback delays on solution quality in case-based reasoning (CBR) have prompted research interest in feedback propagation as an approach to addressing the problem. We argue in this paper that the ability of CBR systems to learn from experience in the absence of immediate feedback is limited by eager commitment to the adaptation paths used to solve previous problems. Moreover, it is this departure from lazy learning in CBR that creates the need for maintenance interventions such as feedback propagation. We also show that adaptation path length has no direct effect on solution quality in many adaptation methods and examine the implications for problem solving and learning in CBR. For such "path invariant" adaptation methods, we demonstrate the effectiveness of a "lazier" approach to learning/problem solving in CBR that avoids commitment to previous adaptation paths and hence the need for feedback propagation.

Keywords: Case-based reasoning, lazy learning, adaptation.

1 Introduction

In case-based reasoning (CBR), a target problem is solved by adapting the solution from the most similar case, or simply by reusing the solution from the most similar case without adaptation [1,2]. It is this *lazy* approach to learning/problem solving that distinguishes CBR from *eager* learning algorithms that create abstractions such as decision trees from training data [3]. Another important feature of CBR is the ability to learn from experience as new cases are added to the case base. However, there is increasing awareness of the effects of feedback delays on solution quality in CBR, and maintenance strategies for addressing this problem have been proposed by several authors [4–6].

For example, Leake and Whitehead [4] investigate several approaches to propagating feedback, when received for a given case, to related and/or similar cases. In one such algorithm, feedback is propagated to all *descendants* of the reference case (defined as those cases that were generated from the reference case by a series of adaptations). Feedback propagation is guided by *case provenance* information captured by the system as each new case is added to the case base (i.e., the set of cases that contributed, directly or indirectly, to the new case's solution). The aim of feedback propagation is to reduce the effects of feedback delays on solution quality. Another

A. Ram and N. Wiratunga (Eds.): ICCBR 2011, LNAI 6880, pp. 151–165, 2011.

problem associated with lack of feedback in CBR is that the solution for a given problem may depend on the order in which previous cases were added to the case base [4].

In this paper, we examine in depth some of the issues brought to light by recent work on case provenance and feedback propagation. The aim of our analysis is to provide a better understanding of the CBR process in the absence of immediate (or any) feedback and its susceptibility to the problems noted by Leake and his co-workers [4–6]. We argue that the ability of CBR systems to learn from experience in the absence of immediate feedback is limited by eager commitment to adaptation paths that determine case provenance but may prove to be sub-optimal in future problem solving. Moreover, it is this departure from lazy learning in CBR that creates the need for maintenance interventions such as feedback propagation. It is also a primary cause of the "order dependence" problem noted by Leake and Whitehead [4].

In previous work, we proposed a "lazier" approach to learning/problem solving in CBR, called Lazier CBR, which uses breadth-first search to discover the shortest possible adaptation path from a seed case (or other case whose solution is known to be correct) to a given problem [7]. The discovered adaptation path is then used to solve the target problem, a process that may involve generating new solutions for some of the cases in the path. In this way, Lazier CBR avoids commitment to previous adaptation paths, and hence the need for feedback propagation. An underlying hypothesis in the approach is that using the shortest available adaptation path may provide more accurate solutions in situations where solution quality tends to deteriorate as the lengths of adaptation paths increase.

However, we show in this paper that many adaptation methods are "path invariant", in the sense that any adaptation path from a given seed case to a target problem gives the same solution as adapting the seed case directly to solve the target problem. An important consequence is that adaptation path length has no direct effect on solution quality for path invariant adaptation methods. Moreover, a common feature of the path invariant adaptation methods we identify is that any case can be adapted to solve a given problem. In this situation, any seed case C provides an adaptation path $C \rightarrow P$ of the shortest possible length that can be used to solve a given problem P.

An alternative to Lazier CBR that we propose in light of this analysis, and show to be effective for a variety of estimation and classification tasks, is an even lazier approach called Lazier$^+$ CBR. In Lazier$^+$ CBR, a target problem is solved by adapting the most similar seed case (or other case whose solution is known to be correct). For path invariant adaptation methods, Lazier$^+$ CBR avoids the computational effort required for adaptation path discovery in Lazier CBR. It is also more efficient than traditional CBR in that cases with unconfirmed solutions play no part in the solution of new problems, and so do not contribute to retrieval effort.

In Sections 2 and 3, we highlight the issues that Lazier$^+$ CBR aims to address, such as the limited ability of CBR systems to learn from experience in the absence of immediate (or any) feedback. In Section 4, we show that path invariance is a property shared by many common adaptation methods, and examine the implications for CBR problem solving and learning in estimation and classification tasks. In Section 5, we examine the hypothesis that for path invariant adaptation methods, a Lazier$^+$ CBR system learns more effectively from experience, in the absence of feedback, than a traditional CBR system. Our conclusions are presented in Section 6.

2 Adaptation Paths in CBR

In this section, we introduce the basic concepts in our analysis and examine the role of adaptation paths in a traditional CBR system. To simplify the discussion, we do not consider CBR approaches in which two or more retrieved cases may contribute directly to the solution of a target problem, for example as in CBR approaches to estimation based on adaptation triples [8].

Seed and Non-Seed Cases. Before a CBR system can begin to solve new problems, it must first be provided with one or more "seed" cases with solutions that are known to be correct. Seed cases are typically provided by a domain expert or imported as legacy cases. We will refer to cases created by the system (i.e., by adapting an existing case to solve a new problem and retaining the problem and its solution as a new case) as "non-seed" cases.

Case = Problem + Solution. For any case C, we will denote by *problem*(C) the problem represented by C. We will denote by *solution*(C) the solution for C that is stored in the case base, whether or not the stored solution is correct.

Adapted Solution. For any problem P and case C that can be adapted to solve P, we will denote by *adapted-solution*(C, P) the solution for P obtained by adapting C to solve P.

Adaptation Path. A sequence of cases $C_1, \ldots, C_n$ provides an adaptation path $C_1 \rightarrow \ldots \rightarrow C_n \rightarrow P$ from a seed case C_1 to a given problem P if C_i can be adapted to solve *problem*(C_{i+1}) for $1 \leq i \leq n - 1$ and C_n can be adapted to solve P. Note that the solution for a given case in an adaptation path that results from previous adaptations in the path may differ from its solution in the case base, which may have originated from a different adaptation path.

A traditional CBR system does not consider all possible adaptation paths that could be used to solve a given problem P. Instead, it retrieves the most similar case C that can be adapted to solve P and uses it to solve the problem. However, if C is a non-seed case then as we show in Theorem 1 there exists a seed case C_1 and adaptation path $C_1 \rightarrow \ldots \rightarrow C_n \rightarrow$ *problem*(C) that was used to solve problem(C). By adapting C to solve P, the system extends the adaptation path that it used to solve *problem*(C) to create a new adaptation path $C_1 \rightarrow \ldots \rightarrow C_n \rightarrow C \rightarrow P$ that now determines the solution for P.

In the proof of Theorem 1, we assume there is no deletion of cases, for example for maintenance purposes [9,10].

Theorem 1. *For any non-seed case C, there exists a seed case C_n and adaptation path $C_n \rightarrow \ldots \rightarrow C_1 \rightarrow$ problem(C) that determines the solution for C.*

Proof. If C is a non-seed case, then *problem*(C) must have been solved by adapting another case C_1, namely the adaptable case that was most similar to *problem*(C) when that problem was presented to the system. If C_1 is a seed case, then the required adaptation path is $C_1 \rightarrow$ *problem*(C). If C_1 is not a seed case, then *problem*(C_1) must have been solved by adapting another case C_2. If C_2 is a seed case, then the required adaptation path is $C_2 \rightarrow C_1 \rightarrow problem(C)$. If C_2 is not a seed case, then we can continue as long as necessary to build a sequence of cases $C_1, \ldots, C_n$ such that for $1 \leq i \leq n - 1$, C_{i+1} is the case that was adapted to solve $problem(C_i)$. Moreover, C_1 was already in the case base before C was created, and C_{i+1} was already in the case base before C_i was created for $1 \leq i \leq n - 1$. It follows that $C, C_1, \ldots, C_n$ must all be distinct cases. We also know that there can only be a finite number of cases in the case base, and at least one of them must be a seed case. So as we continue to extend our sequence of cases, it must eventually be true that the last case in the sequence (C_n) is a seed case. We have now established as required the existence of a seed case C_n and adaptation path $C_n \rightarrow \ldots \rightarrow C_1 \rightarrow problem(C)$ that determines the solution for C. □

3 Why Lazier⁺ CBR?

Fig. 1 shows an example case base and a target problem (P) that we use in this section to highlight the issues that Lazier⁺ CBR aims to address. There are just two attributes in the description of a case with integer values in the range from 0 to 6. Existing cases are numbered in the order in which they were added to the case base and seed cases (C_1, C_7, C_8) are shown as black circles. The adaptation paths used to generate the non-seed cases C_3, C_6, and C_{12} are also shown. For example, the adaptation path that determines the solution for C_6 is $C_1 \rightarrow C_4 \rightarrow C_5 \rightarrow problem(C_6)$.

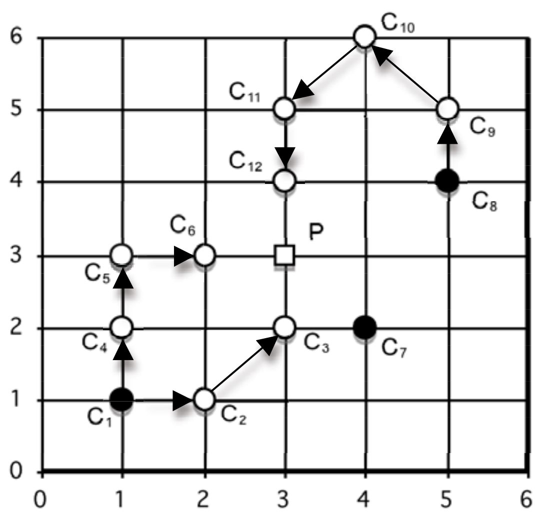

Fig. 1. An example case base in which 3 cases (C_3, C_6, C_{12}) are equally similar to a target problem P

In a traditional CBR system that adapts the most similar case, C_3, C_6, and C_{12} are equally good candidates to be used to solve P. It might be considered that C_3 would be the best choice because it was generated from a seed case by the shortest adaptation path (2 steps). Adapting C_3 to solve P amounts to extending the adaptation path $C_1 \rightarrow C_2 \rightarrow problem(C_3)$ that was used to solve $problem(C_3)$ to create a new adaptation path $C_1 \rightarrow C_2 \rightarrow C_3 \rightarrow P$. There are only 3 steps in the resulting adaptation path as opposed to an adaptation path length of 4 if C_6 is used to solve P or 5 if C_{12} is used to solve P. So using C_3 to solve P can be expected to give better results than C_6 or C_{12} if solution quality is known to deteriorate as the lengths of adaptation paths increase.

However, we show in Section 4 that for many adaptation methods used in estimation and classification tasks, the length of the adaptation path that determines the solution for a given problem has no bearing on solution quality except insofar as problems with longer adaptation paths tend to be less similar to the seed cases from which they were generated. For such "path invariant" adaptation methods, it does not matter whether C_3 or C_6 is adapted to solve P, as both cases will give the same solution. What matters in path invariant adaptation is not the length of the adaptation path, but how similar the target problem is to the seed case that determines its solution. In this context, adapting C_{12} to solve P is likely to give better results than C_3 or C_6 because the seed case from which C_{12} was generated (C_8) is more similar to P than the one from which C_3 and C_6 were generated (C_1).

A traditional CBR system will simply make some arbitrary choice between the equally similar cases C_3, C_6, and C_{12}. It will also ignore the fact that the seed case C_7 is much closer to the target problem than either of the seed cases from which C_3, C_6, and C_{12} were generated. A related issue that CBR researchers have recently begun to consider in the context of delayed/absent feedback is that the solution for a given problem may depend on the order in which previous cases were added to the case base [4].

Definition 1. *The solution that a CBR system provides for a given problem is order dependent if it depends on the order in which cases in existence at the time when the problem is solved were added to the case base.*

For example, if C_7 had been added to the example case base in Fig. 1 before *problem*(C_3) was solved, then the system would have used C_7 instead of C_2 to solve *problem*(C_3). As a result, the adaptation path that determines the solution of C_3 would now be $C_7 \rightarrow problem(C_3)$, which is likely to result in a more accurate solution than $C_1 \rightarrow C_2 \rightarrow problem(C_3)$. Moreover, if the system chooses to solve P by adapting C_3, the adaptation path now used to determine the solution for P would be $C_7 \rightarrow C_3 \rightarrow P$, which is likely to result in a more accurate solution than $C_1 \rightarrow C_2 \rightarrow C_3 \rightarrow P$. With the original ordering of cases, what prevents the system from benefiting from the addition of C_7 as a new seed case is its eager commitment to the adaptation path that it used to solve *problem*(C_3).

In a traditional CBR system, no record is kept of the adaptation paths that determine case provenance. However, the fact remains that these adaptation paths are continually reused by the system to solve new problems, with no attempt to improve or revise them as new cases are created. As the above example illustrates, this eager commitment to previous adaptation paths may limit a CBR system's ability to learn from experience. More generally, if the most similar case C to a target problem P is a seed case, then it is reasonable to expect that adapting C to solve P will provide a good solution for P. However, if C is not a seed case, there may now be several possible adaptation paths from seed cases to C that did not exist at the time when *problem*(C) was solved. Moreover, it is possible that one of these alternative adaptation paths, which the system does not consider, could provide a better solution for C, and could be extended to provide a better solution for P.

In contrast, Lazier$^+$ CBR makes no commitment to adaptation paths used to solve previous problems (which never involve more than a single adaptation step in the approach). Instead, it only allows seed cases, or other cases whose solutions are known to be correct as a result of feedback from a reliable source, to contribute to the solution of new problems. In Lazier$^+$ CBR, the target problem P in Fig. 1 would be solved by adapting C_7, the seed case that is most similar to P. Whether or not the adaptation method is path invariant, this is likely to provide a better solution for P than adapting C_3, C_6, or C_{12}. It also avoids the order dependence problem to which traditional CBR is known to be susceptible [4].

4 Path Invariant Adaptation

In this section, we show that many of the adaptation methods typically used in estimation and classification tasks are path invariant according to the following definition.

Definition 2. *An adaptation method is path invariant if any adaptation path* $C_1 \rightarrow \ldots \rightarrow C_n \rightarrow P$ *from a seed case* C_1 *to a target problem P gives the same solution for P as adapting* C_1 *directly to solve P.*

One example of path invariant adaptation is the approach sometimes referred to as *null* adaptation [11]. In this approach, the solution for the most similar case is reused (i.e., applied to a target problem) without any adaptation. Its use tends to be limited to estimation/classification tasks where the need for adaptation is less critical than in design/configuration tasks [2]. In practice, the most similar case may be required to equal or exceed a predefined similarity threshold before its solution is applied to the target problem without adaptation.

In Section 4.1, we demonstrate the path invariance of two approaches to transformational adaptation for a specific estimation task. In Section 4.2, we consider the implications of path invariant adaptation in a more general context.

4.1 Adapting Soccer Scores

Consider the idea of using a traditional CBR system to estimate the points scored by a soccer team from the numbers of matches it has won and drawn in a series of matches. We used a similar example in previous work on intelligent case authoring [12]. The correct solution for any case C, though unknown to the CBR system, is:

$$points(C) = 3 \times wins(C) + draws(C) \quad (1)$$

In addition to one or more seed cases with correct solutions, the CBR system in our example is provided with the following rules for adapting the solution from the most similar case (C) to solve a given problem P:

Rule 1. **If** $wins(P) > wins(C)$ **then** add $k_1 \times (wins(P) - wins(C))$
Rule 2. **If** $wins(P) < wins(C)$ **then** subtract $k_1 \times (wins(C) - wins(P))$
Rule 3. **If** $draws(P) > draws(C)$ **then** add $k_2 \times (draws(P) - draws(C))$
Rule 4. **If** $draws(P) < draws(C)$ **then** subtract $k_2 \times (draws(C) - draws(P))$

The method used to assess the similarity of a given case C to a target problem P is not important in this discussion, but might for example be based on the Euclidean distance:

$$\sqrt{(wins(P) - wins(C))^2 + (draws(P) - draws(C))^2} \qquad (2)$$

The CBR system solves a target problem by adapting the most similar case. It uses all applicable adaptation rules, with cumulative effect, to adapt the solution for the most similar case. The accuracy of solutions based on the adaptation rules will depend on the values of k_1 and k_2, reflecting the fact that, in practice, such rules may be based on imperfect domain knowledge. For example, Fig. 2 shows a target problem (P) with $wins(P) = 2$ and $draws(P) = 3$. The goal is to estimate $points(P)$, the points scored by a team with these numbers of wins and draws. The most similar case (C) is also shown. Its solution (11) can be seen to be correct from Eqn. 1.

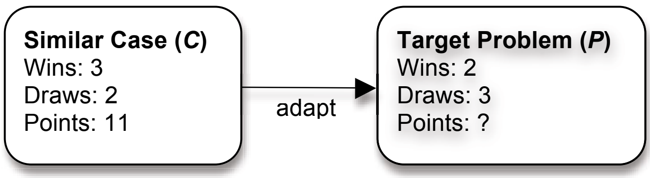

Fig. 2. Adapting the most similar case (C) to solve a target problem (P) in the soccer scores domain

We also know from Eqn. 1 that the correct solution for the target problem is $3 \times 2 + 3 = 9$. However, when Rules 1–4 are applied with $k_1 = 2$ and $k_2 = 1$, the adapted solution for P is $11 - 2 + 1 = 10$. As the solution for the most similar case is correct, it is only the adaptation process that contributes to the error in the solution for P. With $k_1 = 3$ and $k_2 = 1$, Rules 1–4 are guaranteed to give the correct solution for any problem provided the solution for the most similar case is correct. The adaptation rules can be written more concisely as an adaptation *formula* for adapting a case C to solve a given problem P:

$$points(P) = points(C) + k_1 \times (wins(P) - wins(C)) + k_2 \times (draws(P) - draws(C)) \qquad (3)$$

As we show in Theorem 2, adaptation based on Eqn. 3 (or the equivalent rules) is path invariant.

Theorem 2. *In the soccer scores domain, the adaptation formula* $points(P) = points(C) + k_1 \times (wins(P) - wins(C)) + k_2 \times (draws(P) - draws(C))$ *is path invariant for all values of* k_1 *and* k_2.

Proof. For any seed case C_1, problem P, and adaptation path $C_1 \rightarrow \ldots \rightarrow C_n \rightarrow P$, the solution for P obtained by applying the adaptation formula at each step of the adaptation path is:

$points(P) = points(C_n) + k_1 \times (wins(P) - wins(C_n)) + k_2 \times (draws(P) - draws(C_n)) =$

$points(C_{n-1}) + k_1 \times (wins(C_n) - wins(C_{n-1})) + k_2 \times (draws(C_n) - draws(C_{n-1})) + k_1 \times (wins(P) - wins(C_n)) + k_2 \times (draws(P) - draws(C_n)) =$

$points(C_1) + k_1 \times (wins(C_2) - wins(C_1)) + k_2 \times (draws(C_2) - draws(C_1)) + k_1 \times (wins(C_3) - wins(C_2)) + k_2 \times (draws(C_3) - draws(C_2)) + \ldots + k_1 \times (wins(C_n) - wins(C_{n-1})) + k_2 \times (draws(C_n) - draws(C_{n-1})) + k_1 \times (wins(P) - wins(C_n)) + k_2 \times (draws(P) - draws(C_n)) =$

$points(C_1) + k_1 \times (wins(P) - wins(C_1)) + k_2 \times (draws(P) - draws(C_1))$.

That is, the solution provided by any adaptation path from C_1 to P is the same as the solution obtained by adapting C_1 directly to solve P. It follows as required that the adaptation formula is path invariant for all values of k_1 and k_2. □

Fig. 3 shows an example of path invariant adaptation based on Eqn. 3 (or the equivalent adaptation rules) in the soccer scores domain with $k_1 = 2$ and $k_2 = 1$. In this example, adapting Case 1 directly to solve the target problem gives the same solution (16) as adapting Case 1 to solve the problem represented by Case 2 and then adapting Case 2 to solve the target problem.

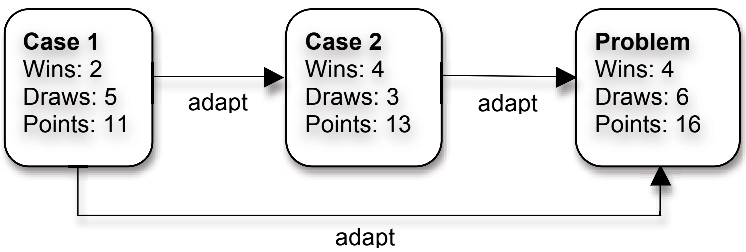

Fig. 3. Path invariant adaptation in the soccer scores domain

Another possible adaptation formula in the soccer scores domain is the following:

$$points(P) = \frac{1 + wins(P)}{1 + wins(C)} \times points(C) \tag{4}$$

Though taking no account of the *draws* attribute, Eqn. 4 captures the idea that points scored can be expected to increase as the number of wins increases. Adding one to $wins(C)$ in the formula avoids the risk of division by zero, and ensures that any case can be adapted to solve a given problem. Also adding one to $wins(P)$ ensures that no adjustment is made when $wins(P) = wins(C)$. When Eqn. 4 is used to adapt the most similar case in Fig. 2, the solution for the target problem is $\frac{1+2}{1+3} \times 11 = 8.25$.

A similar approach to adaptation, though based on attributes with non-zero values, was used by Leake and Whitehead [4] in experiments on the Abalone and Boston Housing datasets from the UCI Machine Learning Repository [13].

Theorem 3. *In the soccer scores domain, the adaptation formula* $points(P) = \frac{1 + wins(P)}{1 + wins(C)} \times points(C)$ *is path invariant.*

Proof. For any seed case C_1, problem P, and adaptation path $C_1 \rightarrow \ldots \rightarrow C_n \rightarrow P$, the solution for P obtained by applying the adaptation formula at each step in the adaptation path is:

$$\frac{1+wins(P)}{1+wins(C_n)} \times points(C_n) = \frac{1+wins(P)}{1+wins(C_n)} \times \frac{1+wins(C_n)}{1+wins(C_{n-1})} \times points(C_{n-1}) =$$
$$\frac{1+wins(P)}{1+wins(C_n)} \times \frac{1+wins(C_n)}{1+wins(C_{n-1})} \times \ldots \times \frac{1+wins(C_2)}{1+wins(C_1)} \times points(C_1) =$$
$$\frac{1+wins(P)}{1+wins(C_1)} \times points(C_1)$$

That is, the solution provided by any adaptation path from C_1 to P is the same as the solution obtained by adapting C_1 directly to solve P. It follows as required that the adaptation formula is path invariant. □

4.2 Path Invariance in General

It might be considered that path invariance is an unusual property of the adaptation methods that we discussed in the soccer scores domain. However, the proof of Theorem 2 can be generalized to show that adaptation is path invariant for any CBR estimation task, numeric attributes $a_1, \ldots, a_r$, coefficients $k_1, \ldots, k_r$, and adaptation formula:

$$adapted\text{-}solution(C, P) = solution(C) + \sum_{i=1}^{r} (k_i \times (\pi_i(P) - \pi_i(C))) \quad (5)$$

where $\pi_i(P)$ and $\pi_i(C)$ are the values of a_i in P and C respectively.

Adaptation is also path invariant for any CBR estimation task and adaptation formula:

$$adapted\text{-}solution(C, P) = \frac{\pi_a(P)}{\pi_a(C)} \times solution(C) \quad (6)$$

where a is a numeric attribute with non-zero values and a direct relationship to the solution attribute, and $\pi_a(P)$ and $\pi_a(C)$ are the values of a in P and C respectively. As previously mentioned, null adaptation [11] is also path invariant.

As discussed in Section 3, the fact that adaptation path length has no direct effect on solution quality for path invariant adaptation methods is one of the motivating factors in our investigation of Lazier$^+$ CBR as an approach to addressing the problems caused by eager commitment to previous adaptation paths in traditional CBR. In Section 5, we examine the hypothesis that for path invariant adaptation methods, a Lazier$^+$ CBR system learns more effectively from experience in the absence of feedback than a traditional CBR system.

5 Empirical Study

In the experiments reported in this section, we assess the ability of a target CBR system to learn from experience by tracking its performance over time as new cases are added to an initially empty case base. The performance measures of interest are percentage accuracy for classification tasks and mean absolute error (MAE) for estimation tasks. We use one or other of these measures to construct a *learning curve* that shows how effectively the system learns from experience. In each experiment, we use a given dataset as a source of seed cases and problems to be solved by the target CBR system. As described in Section 5.1, the proportion of seed cases ($1/r$) in the case base is determined by an integer parameter $r \geq 2$ and remains constant at each of a series of evaluation points. In Sections 5.2 to 5.4, we use this framework to compare the performance of Lazier$^+$ CBR and traditional CBR for a variety of estimation and classification tasks in the absence of feedback.

5.1 Experimental Method

Beginning with an initially empty case base, and a given dataset of size n, we repeat the following steps until the size of the case base reaches $k \times r$, where k is the largest integer such that $k \times r \leq n$.

1. Select an example (description + solution) at random from the dataset and insert it as a seed case into the case base.
2. Remove the selected example from the dataset.
3. Select $r - 1$ examples at random from the dataset and present their descriptions (one at a time) as problems to be solved by a target CBR system. As each problem is solved, add its description and the CBR system's solution to the case base as a new case before the next problem is solved.
4. Remove the examples selected in Step 3 from the dataset.
5. Calculate the percentage accuracy or MAE of the CBR system's (unrevised) solutions over all non-seed cases that are now in the case base.

For example, the system's solution for a non-seed case is deemed to be correct in a classification task if it is the same as the known solution from the dataset. At each of the k evaluation points (Step 5), the proportion of seed cases in the case base is $1/r$. Fig. 4 illustrates our approach to constructing a case base from a given dataset. In this example, $r = 3$ and evaluation points are shown as double vertical lines (||). Seed cases are shown as dark circles. The diagram also shows the adaptation paths that determine the solutions for the first six non-seed cases to be added to the case base in a traditional CBR system. However, this provenance information is not used in our experiments.

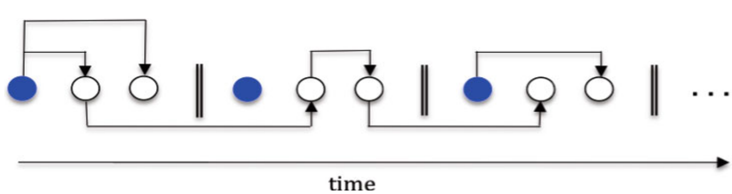

Fig. 4. An example case base incrementally constructed by adding cases to an initially empty case base in groups of $r = 3$ cases

All attributes are equally weighted for the purpose of similarity assessment in our experiments. We define the similarity of two values *x* and *y* of a numeric attribute *a* to be $sim_a(x, y) = 1 - \frac{|x - y|}{max(a) - min(a)}$, where *max*(*a*) and *min*(*a*) are the maximum and minimum values of *a* in the given dataset. We define the similarity of two values of a nominal attribute to be 0 if one or both values are missing. Otherwise, we assign a similarity score of 1 for equal values or 0 for unequal values.

5.2 Estimation in the Soccer Scores Domain

Our first experiment is based on an artificial dataset in the soccer scores domain (Section 4). The dataset contains 169 examples, one for each value of *wins* and *draws* from 0 to 12. The correct solution (*points* = 3 × *wins* + *draws*) is stored with each example description. The resulting dataset provides a source of seed cases and problems to be solved in the construction of a case base and evaluation of a target CBR system as described in Section 5.1. The goal of the CBR system is to estimate the points scored. Adaptation is based in the experiment on the following adaptation formula, which we know to be path invariant from Theorem 2, and in which *C* is the case adapted to solve a target problem *P*.

$$points(P) = points(C) + 2 \times (wins(P) - wins(C)) + 1 \times (draws(P) - draws(C)) \quad (7)$$

Fig. 5 shows the resulting learning curves for $r = 4$ in a traditional CBR system and Lazier$^+$ CBR system. For this value of *r*, the proportion of seed cases at each evaluation point is 1/4. The MAE at each evaluation point is averaged over 100 trials. In Lazier$^+$ CBR, the MAE decreases rapidly to a minimum of 1.2 when the number of seed cases reaches 36. In contrast, it is only after 10 seed cases have been added to the case base that effective learning begins in traditional CBR. Moreover, the lowest MAE achieved by the traditional CBR system is 3.2 compared to 1.2 for the Lazier$^+$ CBR system.

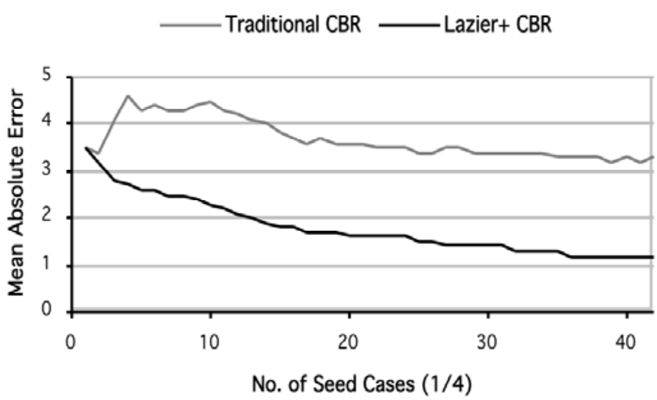

Fig. 5. Learning curves based on mean absolute error (MAE) for traditional CBR and Lazier$^+$ CBR in the soccer scores domain

Table 1. Lowest mean absolute error (MAE) achieved by traditional and Lazier$^+$ CBR systems in the soccer scores domain as the proportion of seed cases increases from 1/10 to 1/2.

	Proportion of Seed Cases (1/*r*)				
	1/10	1/5	1/4	1/3	1/2
Traditional CBR	4.1	3.1	3.2	2.4	1.7
Lazier$^+$ CBR	1.7	1.2	1.2	1.0	0.9

To assess the proportions of seed cases (1/*r*) required for effective learning in traditional CBR and Lazier$^+$ CBR, we repeated the experiment with different values of *r*. Table 1 shows the lowest MAE achieved by traditional CBR and Lazier$^+$ CBR systems as the proportion of seed cases increases from 1/10 to 1/2. The Lazier$^+$ CBR system can be seen to require much fewer seed cases for effective learning than the traditional CBR system. For example, the traditional CBR system requires a proportion of seed cases 5 times greater than the Lazier$^+$ CBR system to achieve a minimum MAE of 1.7.

5.3 Estimating Housing Values

The case base used in our second experiment is constructed from the Boston Housing dataset from the UCI Machine Learning Repository [13]. The dataset contains 506 examples, each representing a residential area in the Boston suburbs described by 13 attributes such as average number of rooms (RM) and distance to employment centers (DIS). The goal is to estimate the median value of owner-occupied homes (MEDV) in a given area. The dataset is used as a source of seed cases and problems to be solved by a target CBR system as described in Section 5.1. All attributes in the dataset are used in the experiment to assess the similarity of a given case *C* to a target problem *P*. Adaptation is based in the experiment on the adaptation formula:

$$\text{MEDV}(P) = \frac{\text{RM}(P)}{\text{RM}(C)} \times \text{MEDV}(C) \qquad (8)$$

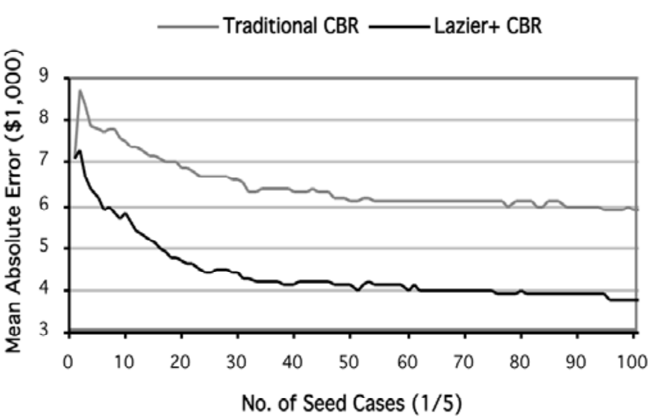

Fig. 6. Learning curves based on mean absolute error (MAE) for traditional CBR and Lazier$^+$ CBR in a case base generated from the Boston Housing dataset

Leake and Whitehead [4] used a similar approach to adaptation in experiments on the Boston Housing and Abalone datasets.

Fig. 6 shows the learning curves for $r = 5$ in a traditional CBR system and Lazier$^+$ CBR system. For this value of r, the proportion of seed cases at each evaluation point is 1/5. The MAE at each evaluation point is averaged over 10 trials. Initially, the Lazier$^+$ CBR system can be seen to learn at a much faster rate than the traditional CBR system. However, both learning curves tend to level off soon after the halfway stage is reached. From this point, the MAE for Lazier$^+$ CBR remains fairly constant at about two thirds of the MAE for traditional CBR. The lowest MAE achieved by the traditional CBR system is 5.9 compared to 3.8 for the Lazier$^+$ CBR system.

5.4 Classification with Null Adaptation

The case base in our final experiment is constructed from the Congressional Voting Records dataset from the UCI Machine Learning Repository [13]. Examples in the dataset represent the votes of 435 US Congressmen on 16 key issues as well as their political affiliations (Democrat/Republican). The goal in this classification task is to predict political affiliation from voting behavior. The dataset is used as a source of seed cases and problems to be solved by a target CBR system as described in Section 5.1. Adaptation is based in the experiment on null adaptation (i.e., the solution for the retrieved case is reused without adaptation) [11].

Fig. 7 shows the learning curves for $r = 3$ in a traditional CBR system and Lazier$^+$ CBR system. For this value of r, the proportion of seed cases at each evaluation point is 1/3. Accuracy levels are shown as percentages averaged over 10 trials. The fastest learning rates in both traditional and Lazier$^+$ CBR can be observed in the early stages of case base growth (i.e., as the number of seed cases increases from 1 to 15). Thereafter, both systems continue to learn at a slower rate until the number of seed cases reaches 100. During this phase, the difference in classification accuracy between the two systems remains fairly constant at around 4% in favor of Lazier$^+$ CBR. The maximum accuracy achieved by the traditional CBR system is 85% compared to 89% for the Lazier$^+$ CBR system. Lazier$^+$ CBR also exhibits a faster learning rate in the early stages of case-base growth than traditional CBR.

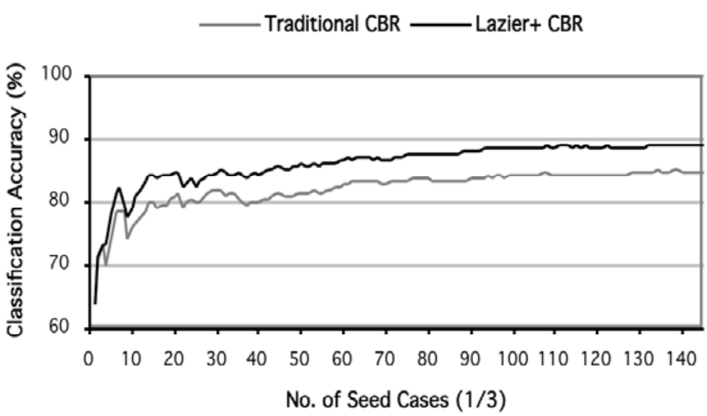

Fig. 7. Learning curves based on classification accuracy for traditional CBR and Lazier$^+$ CBR in a case base generated from the Congressional Voting Records dataset

Like the other results presented in this section, these results support our hypothesis that for path invariant adaptation methods, a Lazier$^+$ CBR system learns more effectively from experience in the absence of feedback than a traditional CBR system.

6 Conclusions

We argued in this paper that the ability of traditional CBR systems to learn from experience in the absence of immediate (or any) feedback is limited by eager commitment to the adaptation paths used to solve previous problems. This departure from lazy learning in CBR is also a primary cause of the order dependence problem brought to light by recent research on case provenance and feedback propagation [4]. We also showed that many adaptation methods used in estimation and classification tasks are path invariant in the sense that any adaptation path from a given seed case to a target problem gives the same solution as adapting the seed case directly to solve the target problem. Importantly, adaptation path length has no direct effect on solution quality for path invariant adaptation methods.

In light of this analysis, we investigated an alternative approach to learning/problem solving in CBR called Lazier$^+$ CBR that avoids commitment to previous adaptation paths, and hence the need for feedback propagation, by allowing only seed cases, or other cases with confirmed solutions, to contribute to the solution of new problems. We also demonstrated the effectiveness of Lazier$^+$ CBR for a variety of estimation and classification tasks based on path invariant adaptation methods. In the estimation/classification tasks that we studied, Lazier$^+$ CBR consistently outperformed traditional CBR in terms of mean absolute error/percentage accuracy. Lazier$^+$ CBR also exhibited faster learning rates and required fewer seed cases for effective learning than traditional CBR. Moreover, Lazier$^+$ CBR is more efficient than traditional CBR because cases with unconfirmed solutions do not contribute to retrieval effort in Lazier$^+$ CBR. In practice, such cases can be stored separately until such time as feedback on their solutions is received.

Investigation of other CBR tasks and adaptation methods that may benefit from Lazier$^+$ CBR, or variations of the basic approach, is an important direction for our future research in this area.

References

1. Aamodt, A., Plaza, E.: Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. Artificial Intelligence Communications 7, 39–59 (1994)
2. López de Mántaras, R., McSherry, D., Bridge, D., Leake, D., Smyth, B., Craw, S., Faltings, B., Maher, M.L., Cox, M.T., Forbus, K., Keane, M., Aamodt, A., Watson, I.: Retrieval, Reuse, Revision and Retention in Case-Based Reasoning. Knowledge Engineering Review 20, 215–240 (2005)
3. Aha, D.W.: The Omnipresence of Case-Based Reasoning in Science and Application. Knowledge-Based Systems 11, 261–273 (1998)

4. Leake, D., Whitehead, M.: Case Provenance: The Value of Remembering Case Sources. In: Weber, R.O., Richter, M.M. (eds.) ICCBR 2007. LNCS (LNAI), vol. 4626, pp. 194–208. Springer, Heidelberg (2007)
5. Leake, D., Dial, S.A.: Using Case Provenance to Propagate Feedback to Cases and Adaptations. In: Althoff, K.-D., Bergmann, R., Minor, M., Hanft, A. (eds.) ECCBR 2008. LNCS (LNAI), vol. 5239, pp. 255–268. Springer, Heidelberg (2008)
6. Leake, D., Kendall-Morwick, J.: External Provenance, Internal Provenance, and Case-Based Reasoning. In: Marling, C. (ed.) ICCBR 2010 Workshop Proceedings. TR-INF-2010-06-03-UNIPMN, pp. 87–94. University of Piemonte Orientale A. Avogadro (2010)
7. McSherry, D.: Towards a Lazier Approach to Problem Solving in Case-Based Reasoning. In: Marling, C. (ed.) ICCBR 2010 Workshop Proceedings. TR-INF-2010-06-03-UNIPMN, pp. 95–101. University of Piemonte Orientale A. Avogadro (2010)
8. McSherry, D.: Demand-Driven Discovery of Adaptation Knowledge. In: 16th International Joint Conference on Artificial Intelligence, pp. 222–227. Morgan Kaufmann, San Francisco (1999)
9. Smyth, B., McKenna, E.: Competence Models and the Maintenance Problem. Computational Intelligence 17, 235–249 (2001)
10. Watson, I.: Applying Case-based Reasoning: Techniques for Enterprise Systems. Morgan Kaufmann, San Francisco (1997)
11. Wilke, W., Bergmann, R.: Techniques and Knowledge used for Adaptation during Case-Based Problem Solving. In: del Pobil, A.P., et al. (eds.) IEA/AIE 1998. LNCS(LNAI), vol. 1416, pp. 497–506. Springer, Heidelberg (1998)
12. McSherry, D.: Automating Case Selection in the Construction of a Case Library. Knowedge Based Systems 13, 133–140 (2000)
13. Frank, A., Asuncion, A.: UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences (2010)