Knowledge-Based Systems 14 (2001) 155–161

Knowledge-Based SYSTEMS

www.elsevier.com/locate/knosys

Competence guided incremental footprint-based retrieval

Barry Smyth*, Elizabeth McKenna

Department of Computer Science, Smart Media Institute, University College Dublin, Belfield, Dublin 4, Ireland

Accepted 2 February 2001

Abstract

Case-based reasoning (CBR) systems solve new problems by retrieving and adapting problem solving experiences stored as cases in a case-base. Success depends largely on the performance of the case retrieval algorithm used. Smyth and McKenna [Lecture Notes in Artificial Intelligence LNAI 1650 (1999) 343–357] have described a novel retrieval technique, called footprint-based retrieval (FBR), which is guided by a model of case competence. FBR as it stands benefits from superior efficiency characteristics and achieves near-optimal competence and quality characteristics. In this paper, we describe a simple but important extension to FBR. Empirically we show that this new algorithm can deliver optimal retrieval performance while at the same time retaining the efficiency benefits of the original FBR method. © 2001 Elsevier Science B.V. All rights reserved.

Keywords: Footprint; Case-based reasoning; Case-based retrieval; Competence models

1. Introduction

The success of any case-based reasoning (CBR) system depends critically on the performance of its retrieval component [2,12]. Smyth and McKenna [15] have previously presented a novel retrieval technique, footprint-based retrieval (FBR), which uses a model of case competence to guide retrieval, and which displays superior efficiency characteristics without significantly compromising retrieval competence or solution quality. However, FBR's efficiency benefits do result in minor competence and quality reductions (typically 1–5% below optimal levels), and since there are always situations where optimal competence and quality must be guaranteed, this means that there are certain situations where FBR, as it stands, is not applicable.

In this paper, we describe a simple but important extension to the FBR algorithm. We demonstrate that this new algorithm (incremental footprint-based retrieval or iFBR) can guarantee optimal retrieval competence and quality while still offering significant efficiency gains. The next sections survey related work on retrieval and outline the competence model that forms the basis of the iFBR method, which itself is described and evaluated in the final sections.

* Corresponding author.

E-mail addresses: barry.smyth@ucd.ie (B. Smyth), elizabeth.mckenna@ucd.ie (E. McKenna).

2. Related work

Retrieval has always received the lion's share of interest from the CBR community. All CBR systems have a retrieval component, and success depends on the efficient retrieval of the right case at the right time. Every retrieval method is the combination of two procedures; a similarity assessment procedure to determine the similarity between a case and target problem, and a procedure for searching the case memory in order to locate the most similar case. In this paper, we are most interested in the latter.

To date research has focused on reducing the search needed to find the best case without degrading competence or quality [3,6,7,11,13,16,18]. The simplest approach is an exhaustive search of the case-base (the brute-force method), but this is not viable for large case-bases. Thus, the basic goal is to avoid the need to examine every case, for example by processing the case data in order to produce an optimised memory structure that facilitates a more directed search procedure.

One approach is to build a decision-tree over the case data (e.g. Ref. [18]). Each node and branch of the tree represents a particular attribute-value combination, and cases with a given set of attribute-values are stored at the leaf nodes. Retrieval is implemented as a directed search through the decision tree. These approaches are efficient but may not be appropriate for case-bases with incomplete case descriptions, or where the relative importance of individual case features can change.

PII: S0950-7051(01)00092-2

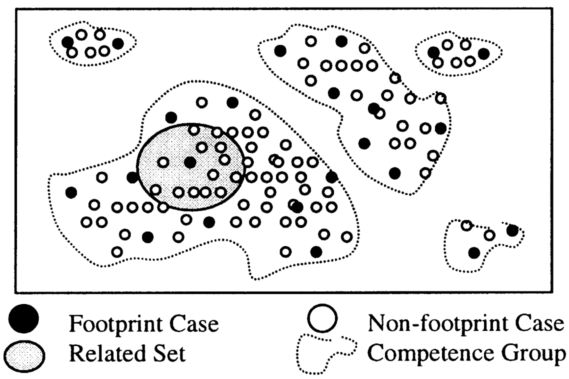

Fig. 1. A sample case-base showing competence groups, footprint cases, and related sets.

Spreading activation methods [3] represent case memory as a 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, however there can be a significant knowledge-engineering cost associated with constructing the activation network. Furthermore, spreading-activation methods may require specific knowledge to guide the spread of activation throughout the network. Related network-based retrieval methods are proposed by Lenz [7] and Wolverton and Hayes-Roth [19].

Perhaps the simplest approach to reduce retrieval cost is to search a *reduced* or *edited* case-base. This strategy is often used to improve the performance of nearest-neighbour techniques by editing training data to remove unnecessary examples (e.g. Refs. [1,4,5]. Many powerful editing strategies have been developed, often preserving competence with a greatly reduced edited case-base.

With the above methods there is a risk in not examining every case during retrieval. The best case may be missed and this can result in a lower quality final solution when compared to the solution that would have been produced by adapting the best case. Even worse, it can mean a problem solving failure if the retrieved case is not close enough to the target for successful adaptation.

3. Model of case competence

In this section, we briefly outline the important components of the competence model used to guide FBR methods (see also Refs. [8,13,15,16,17]).

3.1. Local competence estimates

The local competence contributions of cases are characterised by two sets. The *coverage set* of a *case* is the set of *target problems* that this case can solve, while the *reachability set* of a *target problem* is the set of *cases* that can solve this target. It is not feasible to enumerate all possible future target problems, but by using the case-base (C) itself as a proxy for the target problem space we can efficiently estimate these sets as shown in Defs. (1) and (2).

$$\text{CoverageSet}(c) = \{c' \in C : \text{Solves}(c, c')\}, \tag{1}$$

$$\text{ReachabilitySet}(c) = \{c' \in C : \text{Solves}(c', c)\}. \tag{2}$$

3.2. Shared coverage and competence groups

Coverage and reachability sets provide local competence estimates, but to estimate the global competence of cases it is necessary to model the interactions between their local competences.

$$\text{RelatedSet}(c) = \text{CoverageSet}(c) \cup \text{ReachabilitySet}(c), \tag{3}$$

$$\text{For } c1,\ c2 \in\ C,\ \text{SharedCoverage}(c1, c2)$$
$$\text{iff } [\text{RelatedSet}(c1) \cap \text{RelatedSet}(c2)] \neq \{\}, \tag{4}$$

$$\text{For } G = \{c1, \ldots, cn\} \subseteq C,\ \text{CompetenceGroup(G)iff } \forall ci$$
$$\in G,\ cj \in G - \{ci\} :\ \text{SharedCoverage}(ci, cj)\ \forall ck$$
$$\in C - G, \neg;$$
$$\exists cl \in G :\ \text{SharedCoverage}(ck, cl). \tag{5}$$

The related set of a case is the union of its coverage and reachability sets (Def. (3)). If the related sets of two cases overlap they are said to exhibit *shared coverage*(Def. (4)). Cases can be grouped into so-called *competence groups* which are maximal sets of cases exhibiting shared coverage (Def. (5)). In fact, every case-base can be organised into a unique set of competence groups which, by definition, do not interact from a competence viewpoint — while each case within a given competence group must share coverage with another case in that group, no case from one group can share coverage with any case from another group (Fig. 1).

3.3. Footprint cases and the footprint set

Every group makes a unique (and independent) contribution to competence, but not every case makes a positive competence contribution [13]. The cases that do are called *footprint cases* and the *footprint set* is that minimal set of footprint cases that collectively provides the same coverage as the entire group (see Fig. 1). This set is important because it is only these cases that we need to consider when estimating the competence properties of a given group or case-base (the footprint set of the case-base is the union of the footprint sets of its competence groups).

The footprinting algorithm shown in Algorithm 1 operates by adding cases to the growing footprint set (FP). Each time a case is added, all of the cases that it covers are

```
COV-FP(G)
R-Set ← cases in G
FP ← {}
While R-Set is not empty
  C ← case in R-Set with largest coverage set size
  FP ← FP ∪ {C}
  R-Set ← R-Set - CoverageSet(C)
  Update coverage sets of R-Set cases
EndWhile
Return (FP)
```

Algorithm 1. Computing the footprint set of a Group, G.

removed from the remaining case set (R-Set), and the next case to add is the one with the largest coverage set with respect to the remaining cases (see also Refs. [8,15,16,17]).

4. Incremental FBR

FBR was first introduced by Smyth and McKenna [15]. Its key innovation stems from its use of the above competence model to guide the retrieval process. In this section, we outline the FBR algorithm and show how it can be extended to produce the iFBR method.

4.1. Footprint-based retrieval

FBR is an edited-set retrieval method that uses the above competence model to edit the case-base with respect to a specific target problem. The algorithm (Algorithm 2) has two stages. Stage one identifies the local region of the case-base that contains the target problem. Stage two focuses the search in this region to locate the nearest case to the target. During each step the competence model is used to guide the search in the right way.

Stage 1: Retrieving from the footprint set. During the first stage the target problem is compared to each case in the footprint set of the case-base, in order to locate the case that best matches the target. This case is termed the *reference case* and acts as an index into the case-base for the next stage. The footprint set serves to mark out the principal regions of competence within the case space and thus is a useful structure to guide retrieval. Furthermore, the footprint set contains only a fraction of the case-base and therefore has a low associated search cost.

Stage 2: Retrieving from the related set. The reference case may, or may not, be able to solve the current target, it may even be the closest case to the target in the entire case-base — however, this cannot be guaranteed. This next stage of retrieval compares the target to related (non-footprint) cases in the case-base in order to locate the most similar case in the case-base. The related cases are those that are similar to the reference case, that is, those cases that are elements of its related set. During the second stage of retrieval each of the cases in the reference case's related set is compared to the target, a low-cost operation due to the small sizes of related sets.

4.2. Extending FBR

One of the problems with the FBR method is that there are no guarantees that the best case in the case-base (closest to the target) will be contained within the related set of the reference case, and therefore this best case may not be retrieved. In fact in earlier experiments we have found this problem to occur in between 1 and 5% of retrievals [15]. This is the reason for the minor competence and quality degradation found for the FBR method when compared to the brute-force approach.

The new incremental FBR approach solves this problem by extending the second retrieval stage beyond the related set of a single reference case. So, during stage one, the k best footprint cases are selected to produce a *reference set*. Then stage two searches the union of the related sets of these k cases. This allows the new approach to 'dig deeper' into the case-base in the region of the target and, for large enough values of k, guarantees that the best case is retrieved. By considering multiple reference cases in order of their

```
Target ← Current target problem
CB ← Case-Base, FP ← Footprint Set

FBR(Target, CB, FP)

  Stage 1
  RefCase ← closest case in FP to Target

  Stage 2
  RelSet  ← RelatedSet(RefCase)
  Case    ← closest case in RelSet to Target

  Return(Case)
```

Algorithm 2. The footprint-based retrieval procedure.

```
Target ← Current target problem
CB ← Case-Base, FP ← Footprint Set

iFBR(Target,CB,FP)

  Stage 1
  RefSet ← k nearest FP cases to Target

  Stage 2
  For each c ∈ RefSet
    RelSet←RelSet ∪ RelatedSet(c)
  EndFor

  Case  ← closest case in RelSet to Target

  Return(Case)
```

Algorithm 3. The incremental footprint-based retrieval procedure.

similarity to the target we are effectively guiding the search in the target region. Note that $k = 1$ is the original FBR method and $k = 0$ corresponds to a FBR without stage two (that is, the reference case is retrieved; Algorithm 3).

4.3. Discussion

FBR and iFBR could be viewed as similar to other edited-set methods, as they search a subset of the available cases. However, the subset used by an edited-set technique such as CNN is computed eagerly, at training time, without reference to a specific target problem. In contrast, FBR and iFBR combine cases from a once-off subset of the entire case-base (the footprint set) with cases chosen (lazily) with respect to the target. This allows the FBR approaches to adapt their search space for the current target problem, thereby greatly improving retrieval competence and quality.

5. Experimental studies

We claim that iFBR benefits from superior efficiency and optimal competence and quality characteristics. We support these claims by evaluating three retrieval algorithms: (1) COV-FP, iFBR using the footprinting method described in Algorithm 1; (2) CNN-FP, iFPR using the CNN footprinting method [5]; (3) Standard, a brute-force search of the case-base.

The experiments use two publicly available case bases: 1400 cases from the Travel domain (available from the Case base Archive at http://www.ai-cbr.org, [7]) and a 500 case case-base from the Residential Property domain (available from the UCI Machine Learning Repository). We also produced extra cases for each domain. For the Travel domain 400 duplicate cases and 400 near-miss (redundant) cases are added to generate a total of 2200 cases. For the Property domain 200 duplicates and 200 near-misses are added to give a total of 900 cases. These data-sets are processed to produce 30 different case-bases of size 700 with accompanying (non-overlapping) target problem sets of 200 cases.

We built CBR systems for these case-bases. For each system the solvability criterion was based on a similarity threshold; a target problem was successfully solved if the similarity between it and the retrieved case exceeded the threshold.

5.1. Retrieval efficiency

The first experiment is concerned with evaluating the efficiency of the retrieval algorithms. Efficiency is measured as the inverse number of cases examined during retrieval. This is a fair measure since all four algorithms perform a simple search through a set of cases using the same standard weighted summed similarity operator; in other words there is no difference in the case similarity costs between the different retrieval methods.

Method. Each case-base is tested with respect to its target problem set and the average retrieval cost is computed. For the iFBR methods the value of k is increased until such time as CNN-FP and COV-FP retrieve the same case as the case retrieved by the standard, brute-force approach; This occurs are $k = 3$ and 4 (for COV-FP and CNN-FP) and at $k = 5$

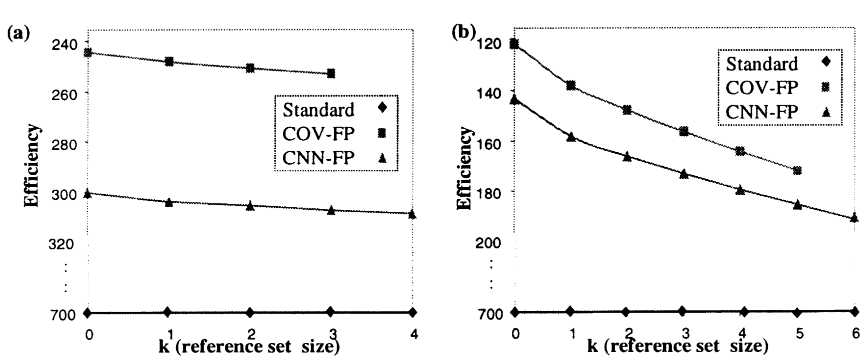

Fig. 2. Retrieval efficiency vs. k (reference set size) for (a) travel, and (b) property domains.

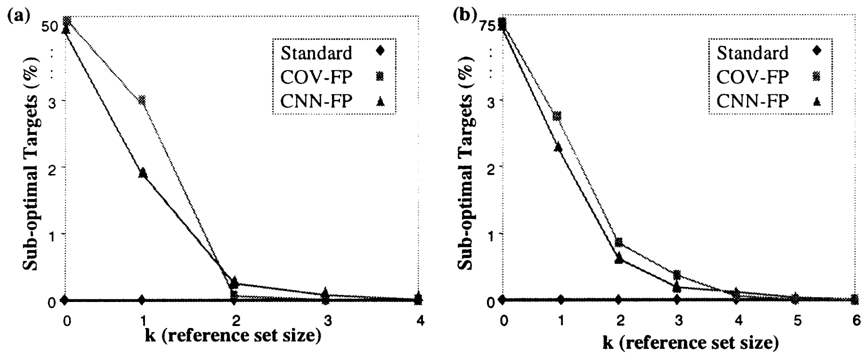

Fig. 3. Retrieval optimality vs. k (reference set size) for (a) travel, and (b) property domains.

and 6 (for COV-FP and CNN-FP), in the Travel and Property domains, respectively.

Results. The results are shown in Fig. 2 as plots of efficiency (in terms of the average number of cases examined by each retrieval method) vs. the reference set size. Notice that at $k = 0$ the CNN-FP and COV-FP methods are equivalent to a search of the footprint sets only (that is no stage 2 retrieval). Similarly, at $k = 1$ the CNN-FP and COV-FP methods are equivalent to the FBR method with its single reference case.

Discussion. Clearly the footprint-based methods significantly out-perform the Standard approach. In the Property domain, Fig. 2(b), for example, the COV-FP version of iFBR is between 4 and 6 times faster that the standard approach for values of k between 0 and 6. As expected, the efficiency of the footprint-based algorithms decreases for increasing values of k because of the larger sets of cases examined during the second stage of retrieval. However, the extra cost remains low with each increment of k. For instance, in the Travel domain, each increment of k contributes about three extra cases to the second stage of retrieval, and thus decreases efficiency by less than 1% for each increment; see also Section 6. In general, the COV-FP variant of iFBR performs consistently and significantly better than the CNN-FP variant since the footprints produced by COV-FP are smaller than those produced by CNN-FP.

In this experiment, we have focused on efficiency for a 700 case case-base. We have carried out similar experiments for different case-base sizes, but due to lack of space we cannot report these results in detail here. However, very briefly, we have found that as the size of the case-base grows, so too does the speed-up achieved by iFBR (with k set for optimal retrievals). For example, in the Travel domain we achieve a speed-up of between 2.1 and 7.3 for case-bases ranging in size from 200 to 1400 cases; similar results have been observed in the Property domain.

5.2. Retrieval optimality

The problem with FBR that motivates iFBR is that it does not guarantee the retrieval of the optimal case, that is, the best case for a given target. In this experiment, we evaluate the optimality characteristics of iFBR and show how it can guarantee the retrieval of an optimal case.

Method. We follow the same experimental method discussed in Section 5.1, but this time, for each domain and algorithm we note the percentage of times that a sub-optimal case is retrieved for a target problem. We do this for increasing values of k until such time as no sub-optimal retrievals occur.

Results. The results are shown in Fig. 3 as a graph of the percentage sub-optimal targets retrieved vs. k. Obviously, the Standard method has 0% sub-optimal retrievals since by definition it retrieves the best case every time (i.e. it achieves 100% optimality). Again note that $k = 0$ and $k = 1$ correspond to the results for the stage-one only and FBR retrieval methods, respectively.

Discussion. As expected the iFBR variants perform poorly at $k = 0$, when only a stage-one retrieval is executed, the equivalent to searching a static edited set only. For example in the Travel domain the COV-FP and CNN-FP variants *fail* to retrieve the best case 56 and 48% of the time, respectively; see Fig. 3(a). At $k = 1$, the standard FBR technique, the COV-FP and CNN-FP variants improve dramatically, with the failures dropping to 3 and 2%, respectively. Very similar results are found in the Property domain as shown in Fig. 3(b). As the value of k increases beyond 1 we see that the iFBR variants quickly converge on an optimal result. For the Travel domain 100% optimality (matching the 0% failures of the Standard approach) is achieved at $k = 3$ for COV-FP and at $k = 4$ for CNN-FP. Similarly, for the Property domain, perfect optimality is achieved at $k = 5$ for COV-FP and $k = 6$ for CNN-FP. These results demonstrate that the additional iFBR stages do manage to actively reduce the number of sub-optimal retrievals, and indeed that 100% optimality can be achieved even for relatively small values of k where significant efficiency benefits are still available; for example, as shown the Property domain in Fig. 3(b), at $k = 4$ for COV-FP, we have reached 100% retrieval optimality by searching just over 20% of the case-base (that is, a speed-up of between 4 and 5 over the

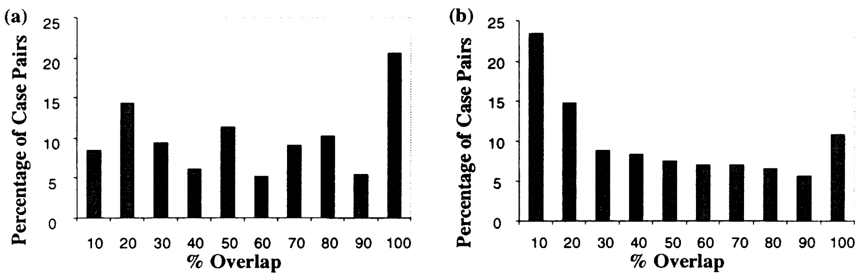

Fig. 4. Related set overlap characteristics for the (a) travel, and (b) property domains.

Standard method). In both domains we find that the COV-FP variant achieves 100% retrieval optimality at lower values of k than CNN-FP, indicating that the COV-FP footprinting technique is producing higher-quality footprint sets than CNN.

6. Discussion

The story so far is that we have introduced a simple but important extension to the successful FBR method. Before concluding we would like to address some of the concerns that have been raised about this extension recently.

6.1. *iFBR results in insignificant gains in retrieval optimality*

FBR on its own benefits from significant efficiency improvements but does suffer from a minor loss in retrieval competence and optimality (typically a 1–5% loss as shown in Ref. [15]). In particular, there are no guarantees that the best available case will be retrieved, as would be the case in a standard brute-force search of the case-base. The point we have made is that sometimes the best available case *must* be retrieved, and in these situations the standard FBR method simply cannot be used. The incremental extension of FBR is a response to this, and although it is a relatively simple extension, we believe it to be an important one because it can be used in situations where FBR cannot; that is, in situations where optimal retrievals must be guaranteed. This point cannot be emphasised enough. The advantage of the new iFBR method is not so much that it leads to large improvements in retrieval optimality — after all, the standard FBR method delivers near optimal (95% + of optimal) retrieval results to begin with. The point is that iFBR can deliver retrieval results that are 100% optimal (for a suitably chosen value of k), and even though the actual increase in optimality works out to be a small percentage, it can nonetheless be an important percentage, and it can make all the difference in domains where optimal retrievals are required.

6.2. *Retrieval time increases by a factor of k when examining k reference cases*

One of the criticisms that has been made against iFBR is that by looking at k reference cases rather than one, we are increasing the retrieval time by a factor of k, and therefore, that iFBR does not scale well with increasing values of k.

This is simply not true. On the face of it, it may appear to be true, because surely examining k related sets (or the k reference cases) will mean examining kn individual cases (where n is the average number of cases in a related set), and hence lead to a factor of k increase in retrieval time? However, the critical point is that nearby reference (footprint) cases will tend to have overlapping related sets, and therefore each new reference case will only contribute a fraction of its related set as new cases to be examined. For example, Fig. 4(a) and (b) show the percentage overlaps between pairs of overlapping related sets in the Travel and Property domain, respectively.

For example, we can see in Fig. 4(a) that over 20% of the pairs of cases share identical related sets in the Travel domain. In fact, for pairs of cases with some overlap between their related sets, the average overlap values are 54 and 40% for the Travel and Property domain, respectively. This means that, in the Travel domain for example, consecutive reference cases will have related sets that overlap by at least 54%, we say 'at least' here because consecutive reference cases are nearby to each other and therefore will have higher than average overlap.

Returning to the efficiency results (Section 5.1) we can see the effect of these overlaps. For example, in the Travel domain Fig. 2(a), the retrieval time for the COV-FP iFBR variant increases by roughly three cases per increment of k; that is, the related set of each new reference case contributes only three new cases to the retrieval process. Similar results are found for the Property domain, with each increment of k contributing between 7 and 10 new cases to the retrieval process (the Property domain has lower overlap characteristics than Travel). Therefore, the additional retrieval cost that comes with iFBR is not as significant as it might seem, and scales well with increasing values of k. Moreover, we argue that in many situations these minor increases in retrieval are acceptable given the benefits associated with the retrieval of an optimal case.

6.3. iFBR requires a competence model that is expensive to maintain

The iFBR method relies on the availability of a comprehensive model of case competence and, of course, there is a cost associated with the construction and update of this model, and because of this there is an additional (and significant) cost associated with iFBR. While it is true that updating the competence model can involve a worst-case cost (of $O(n^2)$ in the size of the case-base), in recent work we have developed a new model building and update procedure that can significantly reduce this cost. In fact, the competence model can be constructed and maintained as a side-effect of iFBR, essentially removing the model-update cost altogether — the same basic computations are carried out in FBR and model update; see Ref. [17] for a complete description and evaluation of this update procedure.

7. Conclusions

We have described a competence-guided approach to retrieval (iFBR) that extends an earlier FBR method [15]. We have evaluated this new approach and demonstrated optimal retrieval results at a greatly reduced retrieval cost. iFBR is a general retrieval technique that makes no assumptions regarding the underlying CBR system, and that is generally applicable to any CBR system once a suitable similarity metric solvability criterion is available. Current and future work will continue to investigate the issue of performance modelling in CBR, and the role of these models in new solutions to problems such as case-base maintenance, case deletion, case-base construction, case-base visualization, and authoring support [8,9,10,14,16,17].

References

[1] D.W. Aha, D. Kibler, M.K. Albert, Instance-based learning algorithms, Machine Learning 6 (1991) 37–66.
[2] A. Agnar, E. Plaza, Case-based reasoning: foundational issues, methodological variations, and system approaches, AI Communications 7 (1) (1994) 39–59.
[3] M.G. Brown, An underlying memory model to support case retrieval, Topics in Case-Based Reasoning, Lecture Notes in Artificial Intelligence, Vol. 837, Springer, Berlin, Heidelberg, New York, 1994, pp. 132–143.
[4] B.V. Dasarathy, Nearest neighbor norms: nn pattern classification techniques, IEEE Press, Los Alamitos, CA, 1991.
[5] P.E. Hart, The condensed nearest neighbor rule, IEEE Transactions on Information Theory 14 (1967) 515–516.
[6] D.B. Leake, A. Kinley, D. Wilson, Case-based similarity assessment: estimating adaptability from experience, in: Proceedings of the 14th National Conference on Artificial Intelligence, AAAI Press, Menlo Park, CA, 1997.
[7] M. Lenz, Applying case retrieval nets to diagnostic tasks in technical domains, Advances in Case-Based Reasoning, Lecture Notes in Artificial Intelligence, Vol. 1168, Springer, Berlin, Heidelberg, New York, 1996, pp. 219–233.
[8] E. McKenna, B. Smyth, Visualising the competence of case-based reasoners, Journal of Applied Intelligence 14 (1) (2001) 95–114.
[9] E. McKenna, B. Smyth, Competence-guided case-base editing techniques, in: Proceedings of the Fifth European Workshop on Case-Based Reasoning, Trento, Italy, 2000.
[10] E. McKenna, B. Smyth, Competence-guided editing methods for lazy learning, in: Proceedings of the 14th European Conference on Artificial Intelligence, Berlin, Germany, 2000.
[11] J.W. Schaaf, Fish and shrink: a next step towards efficient case retrieval in large-scale case-bases, Advances in Case-Based Reasoning, Lecture Notes in Artificial Intelligence, Vol. 1168, Springer, Berlin, Heidelberg, New York, 1996, pp. 362–376.
[12] B. Smyth, M.T. Keane, Adaptation-guided retrieval: questioning the similarity assumption in reasoning, Artificial Intelligence 102 (1998) 249–293.
[13] B. Smyth, M.T. Keane, Remembering to forget: a competence preserving deletion policy for case-based reasoning systems, in: Proceedings of the 14th International Joint Conference on Artificial Intelligence, Morgan Kaufmann, Los Altos, CA, 1995, pp. 377–382.
[14] B. Smyth, E. McKenna, Modelling the competence of case-bases, Advances in Case-Based Reasoning, Lecture Notes in Artificial Intelligence, Vol. 1488, Springer, Berlin, Heidelberg, New York, 1998, pp. 208–220.
[15] B. Smyth, E. McKenna, Footprint-based retrieval, Case-Based Reasoning Research and Development, Lecture Notes in Artificial Intelligence, LNAI 1650, Springer, Berlin, Heidelberg, New York, 1999, pp. 343–357.
[16] B. Smyth, E. McKenna, Building compact competent case-bases, Case-Based Reasoning Research and Development, Lecture Notes in Artificial Intelligence, LNAI 1650, Springer, Berlin, Heidelberg, New York, 1999, pp. 343–357.
[17] B. Smyth, E. McKenna, An efficient and effective procedure for updating a competence model for case-based reasoners, Proceedings of the 11th European Conference on Machine Learning, Barcelona, Spain, 2000.
[18] S. Wess, K-D. Althoff, G. Derwand, Using k–d trees to improve the retrieval step in case-based reasoning, Topics in Case-Based Reasoning, Lecture Notes in Artificial Intelligence, Vol. 837, Springer, Berlin, Heidelberg, New York, 1994, pp. 167–181.
[19] M. Wolverton, B. Hayes-Roth, Retrieving semantically distant analogies with knowledge-directed spreading activation, in: Proceedings of the 12th National Conference on Artificial Intelligence, International Conference on Machine Learning 1992, AAAI Press, Menlo Park, CA, 1994, pp. 56–61.