Competence-Preserving Case-Deletion Strategy for Case-Base Maintenance.

Mohamed-Karim Haouchine, Brigitte Chebel-Morello, Noureddine Zerhouni

Competence-Preserving Case-Deletion Strategy for Case Base Maintenance

Mohamed Karim Haouchine[1], Brigitte Chebel-Morello[1], and Noureddine Zerhouni[1]

[1] Automatic Control and Micro-Mechatronic Systems Department
FEMTO-ST/AS2M–UMR CNRS 6174
24 rue Alain Savary, 25000 Besançon - France
{karim.haouchine, bmorello, zerhouni}@ens2m.fr

Abstract. The main goal of a Case-Based Reasoning (CBR) system is to provide criteria for evaluating the internal behavior and task efficiency of a particular system for a given initial case base and sequence of a solved problems. The choice of Case Base Maintenance (CBM) strategies is driven by the maintainer's performance goals for the system and by constraints on the system's design and the task environment. This paper gives an overview of CBM works and proposes a case deletion strategy based on a competence criterion using a novel approach. The proposed method combines an algorithm with a Competence Metric (CM). Series of tests are conducted using four standard data-sets as well as a locally constructed one, on which, three case base maintenance approaches will be tested and evaluated by competence and performance criteria. Thereafter competence and performance experimental study shows how this method compares favorably to more traditional methods.

Keywords: case-based reasoning, case base maintenance, case base maintenance strategies, competence, performance.

1 Introduction

Maintenance actions are necessary for guaranteeing the good operating in time of an information processing system. From their design stage, Case Base Reasoning (CBR) systems take into account the evolution of their case base and consequently, systems need to be maintained. The CBR cycle contains a determinant phase which consists of the maintenance of the good system's operating. This maintenance is achieved in the learning phase. The latter allows the integration of new cases in the case base in order to improve the system's quality and time response. The learning phase is the step during which the case base is enriched by new resolved problems. Therefore, it is important in this phase to retain only the relevant cases that adhere to the organization of the case base and which complete the system knowledge.

The case base plays a major role which explains the fact that many researches in this area are mainly based on the Case base Maintenance (CBM) [5]. Moreover, CBR knowledge is linked to cases which are affected by all changes in the knowledge sources. The case base is the most sensitive knowledge source to changes in the CBR

system. Its consultation is the most appropriate to set in mention maintenance operations [5]. In this case, maintenance is based on applying update policies of case base representation and is interested to its reorganization in order to facilitate the future reasoning which responds to a set of competence objectives. We have reviewed different research works about CBM and we will present them in section2. However, there are two CBM policies. The first one concerns the case base partitioning and the second deals with the case base optimization. We are especially interested in the latter in order to reach the objectives related to the research time problems as well as the reduction of the case base size while preserving its competence. Section 3 describes a novel method based on the association of a Competence Metric (CM) with cases or instances categorization. This method is positioned within case-deletion strategy framework in which no competence measure has been taken into account. Finally, section 4 describes the obtained results and the comparison with different methods.

2 Case base Maintenance

Case base maintenance implements policies for revising the organization or contents (representation, domain content, accounting information, or implementation) of the case base in order to facilitate future reasoning [5]. Note that this definition considers the information defining an indexing scheme to be an intrinsic organizational component of the case base itself. The CBM involves revising of indexing information, links between cases, and/or other organizational structures and their implementations [12]. The maintenance in CBR involves different operations: out dated, redundant or inconsistent cases may be deleted; groups of cases may be merged to eliminate redundancy and improve reasoning power; cases may be re-described to repair incoherencies [12].

2.1 Case base Maintenance Policies

The CBM approach can be divided in two policies, one concerning optimization and the other a case base partitioning. The objective of these approaches is to reduce the case retrieval time. The optimization policy consists of deleting less relevant cases by following two strategies: addition and the deletion of cases. Whereas, the partitioning policy consists of dividing the case base into several search spaces. This enables to select, in an increasing manner, the attributes which are rich in information and which can cover the structure of the case base [16]. One of the drawbacks of partitioning is during the classification and class selection procedure. When a border element is poorly classified, it is possible to have no answer while it could have been found in the neighbouring class. Several criteria of case base cases were proposed in order to carry out an evaluation concerning case base.

2.2 Criteria for Evaluating Case Base

An "effective" case base is able to answer as many queries as possible efficiently and correctly. The criteria by which one can judge the effectiveness of a case base are given in [9] and [11]. The important criteria that contribute to the evaluation of a case base are: *competence* and *performance*.

- *Competence* is the range of target problems that can be successfully solved.
- *Performance* is the answer time that is necessary to compute a solution for case targets. This measure is bound directly to adaptation and result costs.

Two important competence properties are the coverage set and the reachability set.

- *Coverage* of a case is the set of target problems that it can be used to solve.
- *Reachability* of a target problem is the set of cases that can be used to provide a solution for the target.

Performance depends critically on the accuracy and the cases stored in the case base. Many CBR systems use retrieval methods whose efficiency is related to the case base size, and under these conditions the addition of redundant cases serves only to degrade efficiency by increasing retrieval time [14].

After having given the definitions of the different criteria that permit the evaluation of a case base, we are going to see how they are used and estimated in the different CBM strategies. In the following paragraphs, two strategies will be developed; the addition-case and the deletion-case.

2.3 Case-Addition Strategy

By the successive addition of cases to a originally empty case base, reduced case base will be constructed, thus maximizing criteria. There are two methods, one maximizing the competence criterion, and the other the performance criterion.

Method Maximizing the Criterion of Competence

Smyth and McKenna, which present a method that uses an explicit case competence model based on notions of coverage and reachability. Their "relative coverage" (RC) metric, provides a precise measurement of competence contributions for individual cases. The RC metric, associated with the condensed nearest-neighbour (CNN) algorithm, permits to successively retain only those cases which are not solved by a case that has already been retained, in order to obtain a new reduced case base [14]. This permits the selection of cases which have a big contribution concerning the case base recovery.

Q. Yang and J. Zhu describe a case-addition algorithm for case base compaction that uses a problem-neighbourhood model of case coverage. The Cases based on benefit/usefulness are successively added to the case set retained so far [17].

The interesting part of these methods is the use of models and metrics that makes it possible to guide the case base size reduction by preserving a good competence. The used metrics rank the cases in order to add the most interesting cases in the reduced case base. However, the disadvantage lies in the selection of the cases to add. For

obtaining a reduced case base, the computational time becomes very important. Indeed, for each added case it is necessary to re-examine the whole original case base which can be fastidious.

Method Maximizing the Criterion of Performance

By analogy to the RC metric, Leake and Wilson developed a relative performance (RP) metric aimed at assessing the contribution of a case to the adaptation performance of the system [6]. To attain the benefit of adding the case to the case base, they first assume that the similarity metric will accurately select the most adaptable case for any problem. For each case that might be added to the case base, its contribution was estimated in regard to adaptation performance. The RP value for a case reflects how its contribution to adaptation performance compares to other cases. This metric can be used to guide case addition, favouring cases with low RP values. In the same manner, another metric was developed concerning a "performance benefit" (PB) metric estimating the actual numerical savings that the addition of each case provides. However, on one hand, the RC-CNN method provided a reduction rate of the case base size, better than the PR-CNN and PB-CNN methods. On the other hand, these two previous methods give a result concerning the adaptation cost of the case base cases better than RC-CNN.

2.4 Case-Deletion Strategy

From a given case base, this strategy values cases according to the criteria in order to be able to suppress and bring the case base to a specific number of cases. The evaluation criteria like competence, redundancy and inconsistency, have been used in different methods, which will be explained below.

Suppression Method for using Case base Screening

In this method, the case base will be screened entirely when its size reaches a certain threshold, usually followed by the process of case-deletion.

- *Random Deletion* is a very simple, inexpensive method and completely domain independent. It simply randomly selects and deletes a case from the case base once the case base size exceeds some predefined limit [7].

- *Ironically* is a slightly more complicated method, it calculates the frequency that each case is retrieved and deletes those who are not frequently accessed [8].

- *Deletion based on case base size and density* is a method proposed by B. Smyth and M.T. Keane that studies the case base size, the density and the distribution of cases in a case base. It tries to keep the homogeneity of the cases density [13].

The majority of these methods do not give satisfying results concerning the optimization of the case base size according to the studied criterion. Moreover, there are some methods which are difficult to implement, and those which are easy to implement don't give a convincing results.

Brighton introduced in [2] an Iterative Case Filtering Algorithm (ICF) that iteratively removes a case whose absence produces better results as compared to retaining it. This algorithm also uses coverage and reachability as the selective

criteria. It repeatedly uses a deletion rule that removes cases whose reachability size is greater than that of the coverage until the conditions of the rule are not satisfied.

We are interested particularly in this method because it gives the best results compared to the others [2].

Method from the Cases Categorization

These methods rely on a modeling of the case base competence, proposed by Smyth and Keane [15]. The authors assume that the case-base itself is a sample of the underlying distribution of target problems. A categorization of case in a case base is created according to their competence. The key concepts in categorizing cases are *coverage* and *reachability*.

Given a case base C = $\{c_1,\ldots, c_n\}$ and "*t*" is the set of target cases in the case base. Formally:

- Coverage(c) = {t∈ C: Adaptable (c, t)}, For c∈ C,
- Reachable(c) = {t∈ C: Adaptable (t, c)}, For c∈ C,

As a result, four categories of cases are considered:

- *Pivotal Cases*: a case is pivotal if it is reachable by no other case but itself. Its deletion directly reduces the competence of a system.
  Pivot(s) iff Reachable(c) – {c} = Ø
- *Spanning Cases*: Spanning cases do not directly affect competence. They are so named because their coverage spaces link (or span) regions of the problem space that are independently covered by other cases
  Spanning(s) iff Pivotal(c) ∧ Coverage(c)∩$U_{t\in \text{ Reachable(c)-\{c\}}}$ Coverage(c) ≠ Ø
- *Support Cases*: Support cases are a special class of spanning cases and again do not affect competence directly. They exist in groups, each support providing similar coverage as the others in a group. While the deletion of any one case of a support group does not reduce competence, the removal of the group as a whole is analogous to deleting a pivot, and does reduce competence.
  Support(c) iff ∃ t∈ Reachable(c) – {c}: Coverage(t) ⊂ Coverage(c)
- *Auxiliary Cases*: A case is an auxiliary case if the coverage it provides is subsumed by the coverage of one of its reachable cases. Auxiliary cases do not affect competence at all. Their deletion only reduces the efficiency of the system.
  Auxiliaire(c) iff t∈ Reachable(c) – {c}: Coverage(s) ⊄ Coverage(t)

The methods developed in this area generate a set of target cases so that case bases are categorized. Two basic hypotheses underline these models: case base corresponds to a sample of target or potential cases and space problem is regular, which means that similar problems have similar solutions.

- *Footprint deletion*: this strategy should work to remove irrelevant cases thereby guiding the case base towards an optimal configuration (in the sense that it maximizes competence while minimizing size).

The case categories described above provides a means of ordering cases for deletion in terms of their competence contributions. Auxiliary cases are selected for deletion before support cases, which are chosen before spanning and pivotal cases.

The optimal case base can be constructed from all the pivotal cases plus one case from each support group. This strategy is not designed to eliminate the need for performance-based methods such as utility deletion [11].

- *Footprint Utility deletion*: is the hybrid strategy between footprint deletion and utility deletion. First, the footprint method is used to select candidates for deletion. If there is only one such candidate then it is deleted. However, if there are numbers of candidates, rather than selecting the one with the least coverage or largest reachability set, the candidate with the lowest utility is chosen [11].

The contribution of this paper is in the area of the case deletion strategy. No competence metric to our knowledge has been studied at this level contrary to methods from case addition strategy. Consequently, we propose a methodology of case base optimization by deleting the least quality cases in the case base. This quality of cases is based on the competence measure.

3 Methodology

The proposed methodology is a case deletion strategy method. It is based on a categorization proposed in [11]. It can work using an algorithm associated to the Competence Measure (CM). This methodology deals with two axes. Firstly, the cases are treated. We use Smyth's categorization by guiding the suppression thanks to CM metric, to construct compact competent case base. Secondly, the labeled instances are treated. Using the Smyth categorization, the spanning cases are divided in two sub-categories and the CM metric guides the suppression. The following paragraph will introduce these two contributions.

3.1 Treatment of Cases

The cases are represented by a set of attribute-values. The CM incorporates two properties: coverage and reachability. It gives an individual contribution to the case competence in relation to the size of the latter's coverage set, while attributing to each coverage and reachability case a value that we shall name *coverage value "Vc"* and *reachability value "Vr"*.

$$CompetenceMetric(c) = \frac{Vc(c)}{Vr(c)} \quad (1)$$

Vc(c) = Cardinal of the c case covering set.
Vr(c) = Cardinal of the c case reachability set.

In order to have a case base with good competence, its coverage ratio must be high and its reachability rate must be low. Consequently, the CM is used to guide the deletion of cases in the case base by favouring the cases with a high CM value and deleting those with smaller CM value. Due to this fact, our method consists of reducing the case base size while maintaining a maximal competence.

The case categories will be determined by CM metric. The CM value can be calculated using the Vc and Vr values and therefore lead to the categorization of cases. The properties that allow this categorization are showed in Table 1.

Table 1. Properties of the case categories.

Type of case	Vc(ci)	Vr(ci)	CM(ci)
Auxiliary case	>1	= Vc(ci)	1
Support case group	>1	>1	Same values
Spanning case	≥1	>1	≤1
Pivotal case	1	1	1

The CM value is determinant in the choice of pivotal cases. It is very important that pivotal case is preserved because its deletion reduces directly the competence of a case base. Moreover, a representative that has the highest CM value from each support case group is guarded. On the contrary, auxiliary cases and spanning cases under a certain competence threshold do not affect the competence and can be deleted. It is worth noting that the auxiliary cases are the least important as they make no direct contribution to competence, next are the support cases, then the spanning cases, and finally the pivotal cases. The following is a case base example of with four cases showing the coverage and reachability space of c_1, c_2, c_3 and c_4 cases (Figure 1).

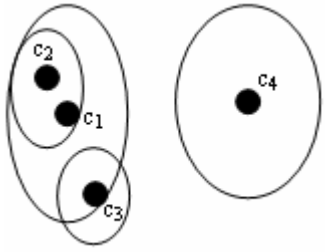

Fig. 1. An example for coverage and reachability

The following results were obtained:
Coverage(c_1)={c_1, c_2, c_3}→ Vc(c1)=3, Reachable (c_1)={c_1, c_2}→Vr(c_1)=2
Coverage(c_2)={c_1, c_2}→ Vc(c_2)=2, Reachable (c_2)={c_1, c_2}→Vr(c_1)=2
Coverage(c_2)={c_1, c_2}→ Vc(c_2)=2, Reachable (c_2)={c_1, c_2}→Vr(c_1)=2
Coverage(c_2)={c_1, c_2}→ Vc(c_2)=2, Reachable (c_2)={c_1, c_2}→ Vr(c_1)=2
CM(c_1) = 1.5, CM(c_2) = 1, CM(c_3) = 0.5, CM(c_4) = 1.

Cases categorization and deletion are determined by the following rules:
Vc(c_2)=Vr(c_2), Vr(c_2) > 1 and CM(c_2)= 1 → auxiliary case → remove case c_2.
Vc(c_3)=1, Vr(c_3)>1 and CM(c_3) < 1 → spanning case → remove case c_3.
Vc(c_4) = Vr(c_4) = CM(c_4) = 1 → pivotal case → retain case c_4.

Therefore we obtained the deletion of two cases (c_2 and c_3) and a case base containing the cases (c1 and c4). By recalculating the CM value of each case, we find that CM(c_1)= CM(c_4) =1 with Va(c_1) = Va(c_4) = 1 et Vr(c_1) = Vr(c_4) = 1.

Therefore we find a reduced case base with two cases forming an optimal case base with two pivotal cases.

3.2 Treatment of Labeled Instances

When we search to treat an instance-base containing instances with their classes, we always need the CM metric and the Smyth categorization. However, one determines two sub-categories in the spanning cases. The first sub-category concerns the inter-class spanning cases and the second, the intra-class spanning cases.

An inter-class spanning case of a given class (for example Class1) is that one, which is partially covered by another case belonging to another class (Class2).

An intra-class spanning case is that one, which is partially covered by another case pertaining to the same class.

In reference to section 2.4, coverage and reachability are based on the characteristic "Adaptable(c,t)". The latter, in the case of classification learning, is defined by the cases retrieval threshold of the case base compared to the target case.

Figure 2 shows an example of a pivotal case ($c_1 \in$ Class2), an inter-class spanning case ($c_2 \in$ Class1), an auxiliary case ($c_3 \in$ Class1), an intra-class spanning case ($c_5 \in$ Class3) and a support group comprising three support cases ($\{c_7, c_8, c_9\} \in$ Class4).

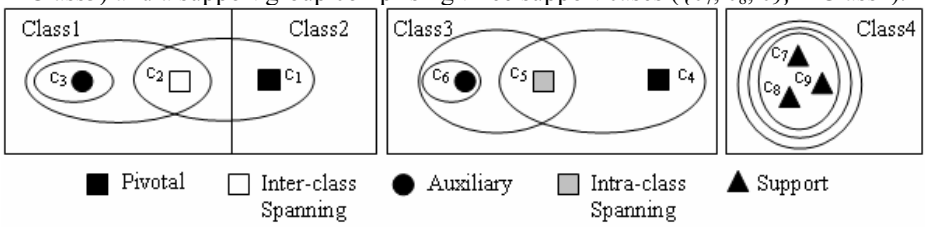

Fig. 2. Case categories

Case 2 belongs to class1 but it is covered by the pivotal case 1 pertaining to class2. This case should not be removed because it contributes to the competence in class1. In addition, a covering threshold is fixed for suppression of the inter-class spanning cases. If the covering of case 1, which is in relation to a case 2 (pertaining to the same class), is higher than a preset threshold, then case 2 is removed, if not, it is kept. This is also valid with the first scenario (treatment of cases). This threshold is given to prevent the removal of cases that have a great contribution of covering and that are reachable by a small part of their covering space. Consequently, the intra-class spanning case definition to be removed is as follows:

Given that $\{c1, c2\} \in$ same class,

- *Intra-class Spanning Case*: Spanning(c_2) iff $\neg$ Pivot(c_2) $\wedge$ Coverage(c_2) $\cap$ $U_{c1 \in (threshold) Reachable(c2)-\{c2\}}$ Coverage (c_2) $\neq \emptyset$

This algorithm concerns the instances-bases containing the labeled instances.

```
For each case to calculate Vc and Vr
  Associate its coverage and reachability set
```

```
Determine the cases categories
   If auxiliary Case then
     Remove all cases
   ElseIf Support Case then
     Classify these cases according to their CM
     values in an increasing way
       For each Support Group Do
         Remove all the case except that which has
         greatest CM value
       EndFor
   ElseIf Intra-class spanning cases then
     Remove all cases except that which has
     "Vc<threshold"
   ElseIf Inter-class spanning cases then
     Retain
   ElseIf Pivotal case then
     Retain
   EndIF
   Stop when each case covers only its own case among
   the existing cases in its class
```

Algorithm 1. Case base Maintenance Algorithm

When the cases are treated, the two sub-categories of the spanning cases are not considered.

4 Experiments

The used method is based on a specific model of competence for case-based reasoning. We argue that it has the potential for guiding the construction of smaller case bases than some existing editing methods without compromising competence, specifically ICF, CNN and CNN with RC distance ordering. In this section we compare the size, the reduction rate, performance and competence of the case bases produced using different editing techniques on a range of standard data-sets. The CNN algorithm was the first reduction technique for the reference base size, based on static considerations [3]. The algorithm aims at reducing the entire input space into a representative subspace with the same properties.

4.1 Assessment of the Proposed Method

Initially, the proposed method is evaluated on case base that relates to an industrial diagnosis dedicated for an e-maintenance platform (SORMEL) [4] (750 cases, 11 attributes and 9 classes). In this latter, the class to be found is an equipment class to be repaired which is formalized in instances form. In the SORMEL case base, the space is taken from the target cases as the total case base space. Firstly, by applying the algorithm, prior to the cases deletion, the following results are obtained:

Table 2. SORMEL Case base Statistics

Pivotal cases	Auxiliary Cases	Inter-class spanning cases	Intra-class spanning cases	Support Cases	Support Group
80	120	50	60	345	95

Concerning the spanning cases, it was listed only in two intra-class spanning cases. After deleting the two auxiliary cases, the two intra-class spanning cases, supports cases and then leaving one support case only for every support group (i.e., 250 supports cases were suppressed), the following statistics are produced:

Table 3. Results obtained after the cases deletion in the SORMEL case base

Initial size of the case base		750
Size of case base obtained		175
Case base performance	Reduction ratio	76,67%
	Accuracy	100%
Competence ratio		100%

The competence rate is of 100% because the obtained case base solves the same number of problems as the initial case base. The obtained results are promising and by applying the proposed method on the Sormel case base, a reduced case base is obtained. Indeed, the obtained case base is considerably reduced by three-quarter (175 per 750 cases), keeping the same initial competence. The results show the very good case base performance which is in relation to the reduction ratio and the accuracy. This good performance is expressed through the decreasing retrieval time with a 100% accuracy. Therefore an optimal case base is obtained.

4.2 Comparative study

This section is divided into two subcategories. The first relates to the comparative study on the competence of the two case addition strategy methods with the proposed method. It should be noted that RC method gives the best results in the case addition strategy. The second relates to the comparison on the performance of the proposed method with the ICF method which has the best results than the other ones of the case deletion strategy.

First Comparative Study (competence)
Three different editing techniques are compared for this experimental study 1) CNN– the standard CNN approach; 2) RC – CNN with cases ordered according to their relative coverage values; 3) CM with cases ordered according to their CM values and the associated algorithm. In order to strengthen the comparison, four different data-sets are used. Travel (351 cases, 34 attributes) and Property (506 cases, 32 attributes) are traditional CBR data-set. The other two, Credit (690 instances, 15 attributes and 2 classes) and Ionosphere (351 instances, 34 attributes and 2 classes) represent classification problems. Property, Credit and Ionosphere data-sets are available from

the UCI Machine Learning Repository (www.ics.uci.edu/~mlearn/ MLRepository .html) [1]. In addition, Travel date-set is available from the AI-CBR Archive (www.ai-cbr.org). In this section, the sizes of the case bases in relation to their competence on unseen target problems are compared. As in [14], each editing strategy is used to generate case bases for the four used data-sets. However, this time 100 random test problems are removed from the training set before the case base construction. The final size of the case bases and their competence over the 100 test problems is noted. The table below (Table 4) illustrates the comparison of the three editing techniques using the four data-sets.

Table 4. A comparison of different editing strategies over the test data-sets in terms of average case base size and competence

Data-set	Property/Method	CNN	RC	CM
Travel	Mean case base size	184.28	197	145.74
	Competence(%)	89.25	88.72	90.84
Property	Mean case base size	55.19	57.81	39.62
	Competence(%)	95.92	95.53	95.91
Credit	Mean case base size	344.84	297.4	215.76
	Competence(%)	58.85	58.95	62.37
Ionosphere	Mean case base size	61.93	46.39	43.87
	Competence(%)	85.78	84.44	86.92

The results are positive. From Table 4, it can be clearly seen that the CM method is more efficient than the other ones by achieving a better cases reduction rate with a finer competence for the four data-sets. The reduction rate given by the developed method is sensibly higher than the one given by the four traditional methods, especially in classification problems represented by the “Credit” and “Ionosphere” data-sets. Concerning the competence value, this one is higher than the corresponding case bases produced by the other methods, though; it is sensibly the same as the traditional method for the “Property” data-set. This shows that our method selects cases that are more competent than those selected by the other methods.

Second Comparative Study (performance)

The second comparative study between the two methods (ICF and CM methods) has been carried out on 18 datasets taken from the UCI repository of machine learning databases [2]. Performance depends on the accuracy and the cases stored in the case base. In our experiments, we retain randomly 20% of the instances for testing and 80% for training. Then the accuracy and resulting size are calculated. This process is repeated several times, the average accuracy and the mean size are given in Table 5.

From the results in this table, several observations can be made. CM method had very good storage reduction and generalization accuracy on average.

Table 5. The classification accuracy and storage requirements for each dataset.

dataset	ICF		CM		The best method
	Storage (%)	Accuracy (%)	Storage (%)	Accuracy (%)	
anneal	22.59	91.35	20.05	100.00	CM
balance-scale	14.67	81.47	13.78	95.83	CM
breast-cancer-l	23.51	72.81	4.02	96.56	CM
breast-cancer-w	4.27	95.14	5.29	93.24	ICF
cleveland	15.60	72.08	6.00	91.01	CM
credit	16.89	82.28	9.30	88.76	CM
glass	31.40	69.64	13.08	72.51	CM
hepatitis	16.33	82.26	11.03	90.03	CM
iris	42.08	92.56	10.66	86.99	CM
lymphography	25.63	77.59	18.92	96.31	CM
mushrooms	12.80	98.64	14.65	98.22	ICF
Pima-indians	17.22	69.17	8.00	93.09	CM
post-operative	7.18	65.28	3.33	83.46	CM
thyroid	21.85	86.63	18.3	86.16	CM
voting	8.88	91.19	2.50	100.00	CM
waveform	18.98	91.19	18.53	96.87	CM
wine	12.00	83.81	3.66	92.94	CM
zoo	52.78	92.42	18.81	100.00	CM
Average	20.25	83.08	10.99	92.33	CM

Some datasets seem to be especially well suited for CM method. For example, it required less than 3% storage for the *voting* and *wine* datasets, yet it achieved even higher generalization accuracy than the ICF method. Generally CM had a higher average generalization accuracy than ICF and also had the lowest storage requirements of the last one. However, ICF is slightly better than CM concerning the accuracy and the storage from the 2 datasets: *breast-cancer-w* and *mushrooms*.

Lastly, the final result (average storage and average accuracy) concerning the 18 datasets shows that the CM method is better than ICF compared to the accuracy (92.33 against 83.08) and almost of the double on storage.

5 Conclusions and Future Work

The suggested method uses an original approach combining an algorithm together with a Competence Metric (CM). This method is used in case bases as well as in instance-bases. The CM metric is established by coverage and reachability notions. The proposed method was evaluated by using two editing methods (CNN and RC-

CNN) and four data-sets. Two of them represent classification problems (Credit and Ionosphere) and the other two are traditional CBR data-set. Also, the introduced method rivals the most successful existing method over 18 domains. The obtained results were positive in terms of case base reduction size, accuracy and best competence. As future work, we plan to develop an auto-increasing method of cases following certain conditions enabling case base maintenance to be continuous.

References

1. Blake, C., Keogh, E., Merz, C.J.: UCI Repository of machine learning databases [http://www.ics.uci.edu/~mlearn /MLRepository.html], University of California, (1998).
2. Brighton, H., Mellish, C.: On the consistency of information filters for lazy learning algorithms, in: Principles of Data Mining and Knowledge Discovery, Third European Conference, PKDD '02, Proceedings, Prague, Czech Republic, September 15–18, (2002).
3. Dasarathy, B.V.: Nearest Neighbor Norms: NN Pattern Classification Techniques, Los Alamitos, California. IEEE Press, (1991).
4. Haouchine, K., Chebel-Morello, B., Zerhouni,N. : Evolution d'un Système de Raisonnement à Partir de Cas Dédié au Diagnostic Industriel, In Plate forme AFIA'2007, 15ème atelier de raisonnement à partir de cas, Grenoble, France, (2007).
5. Leake, D.B.. Wilson, D.C.: Categorizing case-base maintenance: dimensions and directions, Advances in Case-Based Reasoning, 4th European Workshop on Case-Based Reasoning, EWCBR 98, Proceedings, Springer-Verlag, Berlin, 196-207, (1998).
6. Leake, D.B.. Wilson, D.C.: Remembering Why To Remember: Performance-guided case-base maintenance, Advances in Case-Based Reasoning: Proceeding of EWCBR-2K, (2000).
7. Markovitch, S. Scott, P.D.: The Role of Forgetting in Learning, In Proceedings of the Fifth International Conference on Machine Learning, 459-465, (1988).
8. Minton, S.: Qualitative Results Concerning the Utility of Explanation-Based Learning, Artificial Intelligence. 42, 363-391, (1990).
9. Racine, K. and Yang, Q.: On the consistency Management of Large Case Bases: the Case for Validation, In AAAI Technical Report, Verification and Validation Workshop, (1996).
10. Roth-Berghofer, T., Iglezzakis, I.: Six Steps in Case–Based Reasoning: Towards a maintenance methodology for case–based reasoning systems, Includes Proceedings of the 9th German Workshop on CBR, GWCBR, Germany, (2001).
11. Smyth, B., Keane, M.T.: Remembering To Forget: A competence Preserving Deletion Policy for Case-Based Reasoning Systems, In Proceeding of the 14th International Joint Conference on Artificial Intelligent, Morgan-Kaufmann. 377-382, (1995).
12. Smyth, B.: Case-base maintenance, Tasks and Methods in Applied Artificial Intelligence. 11th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Springer-Verlag, Berlin, Germany. 2, 507-516, (1998).
13. Smyth, B., McKenna, E.: Modelling the Competence of Case-Bases, Advances in case-based reasoning, Lecture notes in computer science, Dublin, 1488, 208-220, (1998).
14. Smyth, B., McKenna, E.: Building Compact Competent Case-Bases, Case-based reasoning research and development, Lecture notes in computer science, 1650, 329-342, (1999).
15. Smyth, B., McKenna, E.: Competence models and the maintenance problem, Computational Intelligence: Special Issue on Maintaining Case-Based Reasoning Systems, In Press, (2002).
16. Yang, Q., Wu, J.: Keep it simple: A case-base maintenance policy based on clustering and information theory, 13th Biennial Conference of the Canadian Society for Computational Studies of Intelligence: advances in artificial intelligence, Berlin. 1822, 102-114, (2000).
17. Yang, Q., Zhu, J.: A case addition policy for case-base maintenance, Computational Intelligence Journal, A Special Issue on Case-Base Maintenance, 17, 250-262, (2001).