Informed Case Base Maintenance: A Complexity Profiling Approach

Susan Craw and Stewart Massie and Nirmalie Wiratunga
Computing Technologies Centre, The Robert Gordon University
Aberdeen AB25 1HG, Scotland

Abstract
Knowledge maintenance for Case-Based Reasoning systems is an important knowledge engineering task despite the availability of initial case knowledge and new cases to extend it. For classification systems it is essential that different scenarios for the various classes are well represented and decision boundaries are well defined in the case knowledge. A complexity-based competence metric is proposed that identifies redundant and error-causing cases to be deleted. The metric informs a maintenance tool that enables the engineer to experiment and balance conflicting objectives. Complexity-informed maintenance outperforms benchmark algorithms for redundancy and error reduction tasks.

Introduction
Knowledge acquisition and maintenance are key concerns for knowledge based systems. Case-Based Reasoning (CBR) avoids many of the difficulties associated with the “knowledge acquisition bottleneck” by using cases as independent knowledge fragments in the case base that allow new cases to be retained during problem-solving.

Case knowledge is the fundamental knowledge source for CBR. It is particularly important for classification since exemplars of each class can be stored. Providing sufficient cases are available, CBR’s retrieve and reuse strategy utilises relevant cases. CBR relies on a fundamental assumption: “similar problems have similar solutions”. This is contradicted when classification problems are close to decision boundaries. These areas are an important focus for Error Reduction and Redundancy Reduction algorithms, two important types of knowledge maintenance for case bases (López de Mántaras et al. 2005, §5).

CBR Error Reduction algorithms target cases that may cause problem-solving errors. Here the conflicting objectives are that error-causing cases may be noisy and should be removed, or may be isolated cases or near class boundaries and should be retained. Early research generated Wilson’s (1972) Edited Nearest Neighbour algorithm (ENN) and its successor Tomek’s (1976) RENN. More recent developments focus on more informed selection.

The utility problem – “that the computational benefit from additional knowledge must not outweigh the cost of applying it” – in CBR refers to cases and the added cost of retrieval, and so the case base must be kept “lean and mean”. New cases are not retained automatically during problem-solving, and cases that are no longer useful are removed. Redundancy Reduction algorithms identify cases to remove. Hart’s (1968) CNN added cases incrementally and was the basis for many more informed methods: Aha et al.’s (1991)IB3; Wilson & Martinez’s (2000) Reduction Technique; Brighton & Mellish’s (2001) Iterative Case Filtering; and Delany & Cunningham’s (2004) Conservative Redundancy Reduction. These algorithms balance the conflicting objectives of case base compaction and preservation of competence. The contribution of individual cases must be understood for effective case base maintenance. Smyth & Keane’s (1995) competence model provides insight into the effect of individual cases on competence, and was used to inform Smyth & McKenna’s (2001) footprint algorithms.

Our new complexity-based competence model focuses on interactions at class boundaries in classification domains, ignored by Smyth & McKenna’s model. Our complexity-informed algorithms also give the knowledge engineer the ability to experiment and control the conflicting objectives in case base maintenance. A paper describing complexity-based competence and its application to redundancy reduction was short-listed for best paper award at the 8th European Conference on Case-Based Reasoning (Massie, Craw, & Wiratunga 2006). The use of complexity-based competence has also been applied to case discovery and error reduction (Massie, Craw, & Wiratunga 2005; 2007).

Local Case Complexity
We are interested in the complexity of a case’s neighbourhood by considering the degree of class disagreement. The Case Neighbourhood in Figure 1 illustrates a binary problem with some disagreement in the neighbourhood of C. The Profile consists of bars illustrating the proportion of cases $P_n$ in the neighbouring $n$ cases which disagree with C’s classification. The case complexity is calculated as the average disagreement in the increasing neighbourhoods.

\[
\text{complexity} = \frac{1}{N} \sum_{n=1}^{N} P_n
\]
The calculation of $P_n$ is not weighted using the neighbours’ distances but complexity is biased towards nearest neighbours through repeated inclusion. In experiments the maximum neighbourhood (N) is set to 10 to enable ten nearest neighbours to influence the complexity. A case has complexity 0 if all its neighbours have the same class, the complexity increases if the neighbours have different classes, and isolated outliers have a complexity of one. By looking at the relative complexities of cases we can identify areas of redundancy and noise.

**Complexity Profile**

The complexity measure provides a local indicator of uncertainty within the problem space but the knowledge engineer cannot use this local information directly to gain an insight into the global structure of the case base. Instead we present a profile of case complexities in which the cases are ranked in ascending order of complexity along the horizontal axis and the complexity of each is plotted on the vertical axis. Figure 2 shows a typical profile. Cases with zero complexity appear at the left of the profile and the curve plots the increasing complexity of cases. Three key global case base indicators are illustrated in the case base profile.

**Redundancy:** Where the curve breaks from the axis ($x_1$) is the proportion of cases in single class clusters.

**Error Rate:** Shaded area under the curve corresponds to average case complexity and estimates expected error.

**Noise:** Case complexity > 0.5 means its neighbourhoods contain a majority with different classes. The proportion of potentially noisy cases is the distance to the right of $x_2$.

![Complexity Profile of Case Base](image)

The complexity profile allows us to visualise the overall complexity across a case base but also to make comparisons across different domains. Figure 3 shows the complexity profiles for 3 well-known datasets from the UCI repository (www.ics.uci.edu/~mlearn/MLRepository.html).

**Wine:** The low predicted error rate is confirmed by 10-fold cross-validation 1-NN error (3.7%). The expected noise level is also very low (4%) but there is high redundancy since 75% of cases have zero complexity.

**Hepatitis:** A more gently sloped curve suggests more complex decision boundaries, confirmed by an 18.9% 1-NN error. A moderate level of noise (16%), and several cases with complexity one, suggest isolated cases surrounded by opposing classes. Redundancy is still high (46%).

**Breast Cancer:** The low sloping curve indicates most instances lie close to decision boundaries, confirmed by a 33.9% 1-NN error. High levels of noise (28%) and little redundancy (8%) is distinctive of this dataset.

**Complexity-Guided Knowledge Maintenance**

The case base profile enables the knowledge engineer to view the global impact of local problem-solving and enables the identification of cases that appear to be redundant (complexity 0) or error-causing (high complexity). Thus, case complexity can be used to inform redundancy or error reduction algorithms, and the effect of maintenance can be checked using the profile of the edited case base.

**Redundancy Reduction**

In classification problems, redundant cases are found in clusters of the same class, far from decision boundaries. Case complexity is good at identifying boundary cases with high complexity, and redundant cases with low complexity. Our Complexity Threshold Editing CTE algorithm applies a complexity threshold to remove all cases whose complexity lies below the threshold. Importantly, the knowledge engineer can choose the threshold based on the number of cases and the level of redundancy she wishes to remove. She can also experiment with different thresholds.

The algorithm is applied iteratively with complexities being recalculated after each case deletion to avoid removing all zero complexity cases in a cluster. A $\text{Friend:Enemy}$ distance ratio $F:E_n$ is introduced as a secondary ranking to select the order of deletion for cases with the same complexity.

$$F:E_n = \frac{D_n^F}{D_n^E} = \frac{\sum_{i=1}^{n} d(C, F_i)}{\sum_{i=1}^{n} d(C, E_i)}$$

where $D_n^F$ is the average distance to the $n$ closest friends $F_i$ (same class), $D_n^E$ is the corresponding enemies’ distance (different class), and $d$ is the distance function. Low ratio cases are further from a boundary and are deleted first.

CTE has been evaluated by comparing its redundancy reduction with 3 competitors: Delany & Cunningham’s (2004) CRR, Brighton & Mellish’s (2001) ICF and Smyth & McKenna’s (2001) RC-FP. The 1-NN accuracy and size were measured for edited versions of 5 UCI datasets. Experiments average the results from 10-fold cross-validation experiments repeated ten times.

Figure 4 shows the trade-off between size and accuracy. A description of the Breast Cancer plot (left) explains the diagram. The lowest node corresponds to the original dataset.
with size 100% and accuracy 66%. The next higher node shows the effect of RENN pre-processing to remove noisy cases; this is a normal step before editing. The plot forks here. The solid plot contains the increasingly aggressive CTE thresholding of 0, 0.1, 0.2, 0.3. The dotted plot shows the increasingly aggressive competitors CRR, ICF, RC-FP. Better redundancy reduction algorithms appear higher and only a little to the left because accuracy increases horizontally and size decreases vertically. For Breast Cancer the CTE algorithms (solid) are not as effective as the competitors (dotted). However, the other 4 datasets are typical of more extensive evaluations where CTE outperforms competitors (Massie, Craw, & Wiratunga 2006).

**Error Reduction**

The case complexity and case base profile can be adapted to identify and remove error causing cases (Massie, Craw, & Wiratunga 2007). A version of the F:E, ratio introduced in equation 2 is the basis of a Friend:Enemy distance profile that distinguishes between cases close to decision boundaries and those that may be noisy. The case complexity Pn in equation 1 is replaced by a Friend:Enemy ratio. The Friend-Enemy profile equivalent to Figure 2 gives a global view of noise. Our Threshold Error Reduction algorithm TER applies the same thresholding approach as CTE, but using the Friend:Enemy distance, to remove most complex cases until the accuracy stops improving or the threshold is reached.

TER has been evaluated by comparing its error reduction performance to the ENN and RENN benchmark algorithms (Wilson 1972; Tomek 1976). Figure 5 displays the accuracy-size plots for 3 UCI datasets. Each plot now corresponds to ENN, RENN and TER in sequence above the original dataset. Three versions of each dataset are used: 10% noise corruption (unshaded, dotted), 20% corruption (grey nodes, dotted), and 30% noise (black nodes, solid). Nodes higher and to the right are preferred. Breast Cancer causes a problem for TER also, but for the other datasets TER provides improved compaction and, apart from Hepatitis with 30% noise, improved accuracy.

**Wider Implications**

Case complexity and the resulting case base profile enable an understanding of both the underlying data (cases) and the resulting knowledge (case base). Case complexity is calculated by measuring the irregularity of classification in the neighbourhood of a case. In classification domains, regularity means same class regions of the data, and irregularity is close to classification boundaries or isolated cases. Our methods distinguish between these 3 types of region and visualise the density of each type in the data. We profile the level of regularity or complexity and identify data that lie in areas of regularity or complexity. Indeed boundary regions break the similarity assumption of case-based reasoning: that “similar problems will have similar solutions”.

In CBR systems complexity-based data analysis coincides with an understanding of the key knowledge source, the case base, and so we have presented the work as underpinning knowledge refinement for case base maintenance. Indeed
the same complexity metric has also been used for case discovery and selection (Massie, Craw, & Wiratunga 2005). However, data profiling could be more generally applicable to learned models for classification. Decision tree learners could be presented with a selection of data in which complex areas are strengthened and regular areas are sampled. Learners would focus on areas of complexity in a similar fashion to boosting. The complexity profiles from neighbourhoods may provide valuable insights into the problem space even when the learning algorithms are not explicitly based on nearest neighbour retrieval.

Our research has concentrated on complexity profiles for classification. Recent work by others on case-based regression has concentrated on finding areas of the case base with similar linear-weighted linear regression models (McDonnell & Cunningham 2006). Here, understanding of regularities in the problem-solving space is used to identify pairs of cases whose difference is useful to adapt the retrieved solution for a new problem. It is clear that areas of low complexity, as evidenced by accurate local linear models, are used to improve problem-solving.

An interactive complexity-informed maintenance tool (Figure 6) enables the knowledge engineer to experiment with redundancy reduction and to achieve a compromise between a smaller case base and greater accuracy. The right panel contains parameter selections and a display of the complexity profile (lower curve), threshold (horizontal line) and resulting profile after editing (upper curve). A spring model visualisation shades the redundant cases to be removed. Similar interfaces exist for case discovery and error reduction (Massie, Craw, & Wiratunga 2005; 2007).

![Figure 6: Interactive Redundancy Tool (CTE)](image)

**Conclusions**

Case complexity enables varying levels of aggressiveness in redundancy and error reduction maintenance algorithms. The simpler complexity metric averages the proportion of enemies in increasing neighbourhoods of a case to identify redundant cases with complexities close to zero. The more complex friend/enemy metric calculates the balancing distances to neighbours of the same and different classes to identify error-causing cases. Different thresholds enable the knowledge engineer to control the aggressiveness.

The case base profile allows the knowledge engineer to inspect the level of redundancy, noise and predicted error in the cases. The profile provides a visualisation of the competence of the case base and allows her to experiment with the complexity thresholds of the maintenance algorithms. The profiles for before and after editing enable a view of the effect of the maintenance.

Access to local case complexity and a global complexity profile provides an understanding of the case base that is useful for maintenance. This approach offers the opportunity to visualise datasets for other learned models and to provide informed data cleaning and pre-processing tools.

**Acknowledgments**

We thank Barry Smyth for useful discussions on complexity-informed maintenance that focused the issues in this paper.

**References**


