Abstract. In the field of case based classification little research has so far been done into the step of preselecting possible candidate solutions from a given set of cases. Such a preselection becomes increasingly necessary and profitable as the casebases examined grow larger. This paper contains a description of various preselection strategies and their theoretical comparison as to applicability, pre-computation effort, running time, and error rate (completeness). The approaches are then evaluated in the case based classification shell CcC+, using a practical application on plant classification with appr. 3000 cases. The evaluation also studies these approaches in relation to changes in casebase size.

1 Introduction

When selecting a case similar to the case in question from a large casebase, it is useful to integrate existing domain knowledge in the similarity match. In complex domains, however, this renders the individual steps in the similarity match very expensive. For practical use of large casebases, an examination of all cases would cost much effort. Therefore an essential step within the paradigm of case based reasoning consists in the preselection of potentially similar cases. This step aims at reducing as far as possible the number of cases to be examined in a subsequent similarity match without eliminating any relevant cases.

To our knowledge, practical handling of large casebases in case based reasoning has hardly been dealt with in relevant publications. The evaluation by CASEY [2], for example, was limited to a casebase of 240 cases. It revealed, for one thing, that a much larger variation (number) of cases was required to yield good results. Furthermore, in spite of a selection based on indexation in the hierarchically organised case memory [6] the search effort for these 240 cases was relatively high and increased in an linear fashion with the number of cases. A problem which often arises in evaluating smaller casebases consists in the difference between a preselection and a complete comparison not being noticeable at all.

The difficulties in such a practical evaluation lie with the fact that for a prototype application only very few cases exist and case acquisition involves much effort. In order to evaluate the methods described here on a suitably extensive casebase, appr. 3000 cases were generated from domain knowledge for a heuristic problem solver and random checked for plausibility by an expert.

There are several approaches trying to tackle the problem of preselection. Almost all of them develop selection criteria - sometimes simple ones, sometimes sophisticated ones - which allow for an efficient similarity-based search in the casebase by supplying suitable access structures. For reasons of efficiency, these selection criteria must, of course, be less specific than the subsequent similarity criteria. The difficulty in making use of such less specific though more efficient selection criteria is to ensure that the most similar cases in terms of the similarity measure are amongst the preselected cases.

This problem does not arise with approaches which are built on massive parallel processes (e.g. [7]) and which, in similarity matches, employ a processor for each case in the casebase. Strictly speaking, however, such methods do not select cases at all but carry out a complete similarity match facilitated by extensive hardware facilities. These approaches are therefore not dealt with in the following.

Criteria for efficient preselection depend strongly on the type of similarity measure chosen. Two classes of similarity measures can be distinguished:

1. Hierarchical similarity measures: Supported by a knowledge intensive model of the application domain, hypothetical explanations of the problem are inferred. The various explanation paths inferred are compared to those stored with the cases in the casebase. The better the explanation paths of a compared case tally with those of the new case, the more similar the stored case and the new case are. Similarity measures of this kind are used e.g. by CASEY [8], PROTOS [3], and CREEK [1].
2. Numerical similarity measures: By means of data abstraction, systems belonging to this category such as PATDEX [14] and \text{CcC} (\text{\cite{11}, \cite{5}}) also expand the problem by reliably inferrable links. However, no further explanations are inferred. Instead, for all features appearing in the expanded problem partial similarity is computed in relation to the features of the existing cases used for comparison. The similarity measure for two cases is then computed from the partial similarities of the features occurring in the cases.

In terms of preselection, the inference of explanation paths also is an indexation on the casebase; due to the knowledge intensive model it is, however, very expensive and very difficult to implement with large casebases. Comparing the explanation paths includes a numerical similarity measure but takes place on an abstract level.

Our target being an efficient case comparison based problem solution, the approach presented below appears more promising. It directly defines a numerical similarity measure for the features. Cases may here be considered as vectors in an $M$-dimensional search space, with $M$ equalling the number of features existing in the domain. The case based approach aims at finding the vector closest to the search vector given by the problem.

The vectors in the search space thus opened must subsequently be compared to the search vector in a detailed similarity match. With the help of various preselection strategies, the size of the search space may be reduced. Several such methods applicable in the field of case based classification are depicted in the following. The conditions of the individual methods are presented irrespective of the underlying numerical similarity measure, and problems arising in case of non-fulfilment of these conditions are discussed. The strategies implemented in the case based classification shell \text{CcC} (\text{\cite{5}}) are then evaluated by using them in a plant classification application.

2 Major Preselection Strategies for Case Based Classification

This chapter presents the functioning of the most prominent preselection strategies for case based classification, including standard procedures such as the preselection of weighted features (generalisation of database indices), K-D-trees, hierarchical organisation of the casebase, and domain-specific methods (e.g. other problem solvers). A comparatively new method is called hill-climbing via case neighbours. The description of these procedures pinpoints, in particular, the required access structures:

2.1 Preselection by Means of Weighted Features

In this fairly simple method a certain number of relevant features are picked from the new case, the relevance of a feature increasing with its value-dependent weight as defined by the similarity measure. Around these relevant features $m_i$ a similarity interval $[m_u, m_o]$ is created so that \[ V m_k \in [m_u, m_o] | sim(m_k, m_i) \geq X \] applies for a given minimum similarity $X$. For preselection, all cases found within these similarity intervals are picked from the casebase. The weights of all features where minimum similarity exists are then added up and the selected cases are sorted according to the total sum of weights. A certain number of best cases are subsequently returned as potential candidate solutions. For $N$ cases, the effort of this methods amounts to $O(N)$ value comparisons for searching and $O(N' \cdot \log(N'))$ value comparisons for sorting, with $N'$ being the number of cases returned. This number depends on the predefined minimum similarity. $N'$ normally being considerably smaller than $N$, the increase in effort is roughly linear to that in the number of cases in the casebase. This method can be improved by feature values indices contained in the database in which the cases are stored. However, when working with database indices, it is not possible to use any further domain-specific knowledge. The advantage of this method lies with the fact that no pre-computation is necessary.

2.2 Preselection by Means of Hill-climbing via Case Neighbours

For this variant, a certain number of most similar cases in the casebase are pre-computed and stored for each case. Neighbours may be fully pre-computed which leads to an effort of $N \cdot (N - 1)$ case comparisons for initial computation of all neighbours and to an effort of $2 \cdot (N - 1)$ case comparisons for incremental insertion of a case. The incremental insertion effort, however, may be reduced by means of a simplified heuristic method which computes the neighbours of the new case and enters the new case with the neighbours if it features stronger similarity than the existing neighbours. Case comparison effort is thus reduced by half. If case insertion is preceded by a case similarity match anyway, the results may be re-used, thus avoiding additional effort altogether. To keep the error rate low, occasional exact re-computation may be useful.