Explanation Support for the Case-Based Reasoning Tool myCBR

Daniel Bahls and Thomas Roth-Berghofer
German Research Center for Artificial Intelligence DFKI GmbH
Trippstadter Straße 122, 67663 Kaiserslautern, Germany, and
Department of Computer Science, University of Kaiserslautern,
P.O. Box 3049, 67653 Kaiserslautern
{daniel.bahls,thomas.roth-berghofer}@dfki.de

Abstract
Case-Based Reasoning, in short, is the process of solving new problems based on solutions of similar past problems, much like humans solve many problems. myCBR, an extension of the ontology editor Protégé, provides such similarity-based retrieval functionality. Moreover, the user is supported in modelling appropriate similarity measures by forward and backward explanations.

Case-Based Reasoning
Case-Based Reasoning (CBR), according to (Aamodt & Plaza 1994), basically follows this pattern: One formulates a problem as a query case and the repository of already experienced problem and solution pairs (the case base) will be ordered by similarity to the given query. The most similar cases are used to generate the solution for the posed problem. After a solution is retrieved, the new case (consisting of the new problem and the retrieved solution) is stored in the case base. This new experience can be used in the next retrieval. The CBR system learns.

CBR systems' knowledge can be divided in four knowledge containers (Richter 1995):

• Vocabulary This knowledge container is the basis for the three other containers. It defines attributes and classes for query and case descriptions. In object-oriented CBR systems the vocabulary consists of numerical, symbolic, plain text, and instance type attributes.

• Case Base This is the collection of previously experienced cases (traditional view) or products.

• Similarity Measure The degree of similarity between a query and a case is defined by metrics. Local similarity measures define similarities for each attribute. Global similarity measures, e.g., weighted sum, minimum, or maximum, aggregate the local similarity measures into one similarity value on each class level.

• Adaptation Rules This container provides knowledge for adapting the solution of a case to fit the query. This is often realised with rules.

Adaptation rules are outside the scope of this work. We concentrate on the support of similarity measure modelling.

myCBR\(^1\) is an open-source plug-in for the open-source ontology editor Protégé\(^2\). It follows in the footsteps of the integrated, Smalltalk-based CBR shell CBR-Works (Schulz 1999) with its rich point-and-click user interface. We also implemented a basic export interface to exchange similarity measures with jColibri, a powerful Java-based CBR framework\(^3\), which allows more complex reasoning.

In Protégé users define classes and attributes in an object-oriented way. Protégé also manages instances of these classes, which we interpret as cases. So vocabulary and case base knowledge containers are already handled by Protégé.

The Open-Source Tool myCBR

Figure 1: Similarity measure editors of myCBR

The myCBR plug-in adds several similarity measure editors, which can be applied to the classes and attributes of an ontology. Its retrieval engine finds similar cases for a specified query. Additionally, CSV files can be imported for which a simple similarity model is built automatically if none exists. A standalone retrieval engine allows for easy integration into other applications. Figure 1 shows a screenshot of some of the available editors.

\(^1\)http://mycbr-project.net
\(^2\)http://protege.stanford.edu/
\(^3\)http://gaia.fdi.ucm.es/projects/jcolibri/
For symbolic attribute types similarity values for all possible attribute values are defined using a table. The columns are headed by the symbols defined in the ontology, as are the rows. The similarity value for a query \( q \) and a case \( c \) can now be found in row \( q \) at column \( c \) (see lower half of Figure 1).

This works fine for a small set of symbols. Otherwise, the effort of setting up and maintaining such a table becomes cumbersome, since the amount of entries increases quadratically with the number of symbols. Therefore, another editing mode, based on taxonomies, is available (see right half of Figure 1). One can build up a tree structure upon the symbols where the distance between symbols indicate their similarity.

To define similarity measures for numerical attributes, some simplification is helpful. In order to offer an easy-to-use interface (see left half of Figure 1), we reduce the dimensionality of the similarity function \( \text{sim}(q, c) \) by either calculating the quotient \( q/c \) or the difference \( q - c \). This value is now the parameter for a helper function \( h : D \rightarrow [0, 1] \), which graph is editable by pointing and clicking. Note that the domain \( D \) has the range \([\text{min} - \text{max}, \text{max} - \text{min}]\) for the difference mode, and it has the range \([\text{min}/\text{max}, \text{max}/\text{min}]\) for the quotient mode, where \( \text{min} \) and \( \text{max} \) are the range limits of the numerical attribute. Obviously, the quotient mode is applicable for numerical attributes with a value range not containing zero.

This is just a selection of available local similarity measures, for which explanation support exists.

**Explanation Support in myCBR**

Explanations, in principle, are answers to questions. In this paper we concentrate on questions the knowledge engineer might have during similarity measure modelling (see (Roth-Berghofer 2004) for more details on explanation sources in CBR systems). myCBR provides two kinds of explanations: forward and backward explanations (Richter & Roth-Berghofer 2007). *Forward explanations* explain indirectly by showing different ways to optimise a given result. They open up possibilities for the exploratory use of a device or application. *Backward explanations* explain the result of a process and how it was obtained. Here, we provide a way to understand the results of myCBR’s similarity calculation (backward explanations) and to explore the case base contents (forward explanations).

**Backward Explanations**

After the vocabulary has been set up, some cases have been injected into the case base, and similarity measures have been defined, the CBR system is ready for retrieval. For a query the system delivers a ranking of the case base. But since the model may be complex, the retrieval result may be quite surprising and need some explanation. To increase transparency, myCBR creates an explanation object for each case during the retrieval. This tree-like data structure stores global and local similarity values as comments for each attribute. These retrieval details are presented to the user by tool tips.

Another valuable feature is the option to find the first most similar cases with respect to a single attribute in this rank-