Using Symbolic Similarity to Explain Case-based Reasoning in Classification Tasks

Eva Armengol and **Enric Plaza**
IIIA - Artificial Intelligence Research Institute
CSIC - Spanish Council for Scientific Research
Campus UAB, 08193 Bellaterra, Catalonia (Spain)
Tel.: +1 222 333 4444; Fax: +1 222 333 0000;
E-mail: eva@iiia.csic.es, enric@iiia.csic.es

Abstract

The explanation of the results is a key point of automatic problem solvers. CBR systems solve a new problem by assessing its similarity with already solved cases and they commonly show the user the set of cases that have been assessed as the most similar to the new problem. Using the notion of symbolic similarity, our proposal is to show the user a symbolic description that makes explicit what the new problem has in common with the retrieved cases. Specifically, we use the notion of anti-unification (least general generalization) to build symbolic similarity descriptions. We also present an explanation scheme using anti-unification for CBR in classification tasks that focuses on explaining what is shared between the current problem and the retrieved cases that belong to different classes.

Introduction

Explaining the results of automated problem solving systems is a key issue concerning their acceptability and understandability. These explanations have to support the user in both the understanding of the result and the process to reach it. When this process is not clearly explained in a convincing way the user could reject using the problem solving system. Case-based reasoning (CBR) systems predicts the solution of a problem based on the similarity between this problem and already solved cases. Clearly, the key point is the measure used to assess the similarity among the cases. Sometimes the resulting similarity value is difficult to explain, thus CBR systems commonly show the user the retrieved cases (the set of cases that have been assessed as the most similar to the new problem). Nevertheless, when the cases have a complex structure, simply showing the most similar cases to the user may not be enough.

In this paper we will propose a way to summarize the similarity of a new case with the retrieved cases by means of a symbolic description. These descriptions capture in a symbolic way those aspects of the retrieved cases that are similar to current case; these symbolic similarity descriptions will be showed to the user as an explanation of the CBR prediction for the current case.

In our experience, we observed that for classification problems using the *k-NN* algorithm sometimes the k retrieved cases are classified in different classes. Commonly, these situations are solved using criteria such as the *majority rule* (i.e. the new problem is classified in the same class as the majority of the retrieved cases) to give the classification of the new problem. Nevertheless, this situation needs to be well explained to the user, specially when the majority is not overwhelming (for instance, when $k = 5$ and three of the retrieved cases are in a class $C1$ and the other two cases are in another class $C2$). The approach in this paper is that, in addition to make explicit the similarities of the current case with the retrieved cases, it is useful to make explicit the similarities among the new problem and the retrieved cases in each class separately.

While CBR systems customarily use numeric assessment techniques (usually metrics or distance measures), the key notion in our approach is that of *symbolic similarity*. Using numeric assessment techniques makes sense to induce a ranking on the degree of importance that past cases have with respect to the current case, and finally selecting the higher ranking ones as the set of *retrieved cases*. In order to provide explanations, however, numeric values provide less leverage than symbolic explanations. Thus, we can consider the notion of *symbolic similarity* between two cases as a type of description that expresses aspects common to (or shared by) these two cases.

In fact, taking the notion of generalization from Machine Learning, we can see that any *generalization* of two cases is a description of *some* aspects they both have in common; in other words, a *symbolic similarity* description can be built by any generalization process. The difference is that generalizations in inductive ML are built to symbolically describe necessary (and often also sufficient) conditions for a case to belong to a class; since many generalizations can be built, inductive ML can then be seen as a search process in the space of generalizations. Moreover, inductive ML techniques often focus on finding discriminant generalizations, i.e. general descriptions that predict a case to be of a specific class and not of any other.

In our situation, however, is different: CBR already provides a way to predict a solution. We consider the generalization of two or more cases (e.g. the current case and one or more retrieved cases) as a description of what is simi-

lar, what is shared, among them. Moreover, we will not be building discriminant descriptions, instead we will use generalizations as explanations of the current CBR prediction being endorsed by the currently retrieved cases. Therefore, although a *symbolic similarity* description is technically a *generalization*, the use we put them to is different than the use they have in inductive ML. For this reason, and to avoid confusion with ML usage, we will call the explanations we will build symbolic similarity descriptions.

Approach

The goal of our approach is to explain the CBR result in a way understandable by an user that is expert in some domain but that he has not necessarily be aware of the formalism used to represent the domain. In the present paper we propose an explanation scheme for classification problems that follows the same idea of the symbolic similarity introduced in (Plaza, Armengol, & Ontañón 2005) but that is independent of the CBR method used to solve the problem. Our hypothesis is that the result of the retrieval process is a *retrieval set* C with the k cases; our approach is independent of how the cases in retrieval set are determined, although in the rest of the paper we will assume they are the most similar cases to the new problem following some k-NN technique.

The explanation scheme we propose is based on applying to the set C the concept of *least general generalization*, commonly used in Machine Learning. The relation *more general than* ($\geq_g$) forms a lattice over a generalization space $\mathcal{G}$. Using the relation $\geq_g$ we can define the *least general generalization* or *anti-unification* of a collection of descriptions (either generalizations or cases) as follows:

- $AU(d_1, ..., d_k) = g$ such that $g \geq_g d_1 \wedge ... \wedge g \geq_g d_k$ and does not exists $g' \geq_g d_1 \wedge ... \wedge g' \geq_g d_k$ such that $g >_g g'$

That is to say, g is the most specific generalization of all those generalizations that cover all the descriptions $d_1, ..., d_k$. The interpretation of anti-unification from the point of view of symbolic similarity is the following: consider the anti-unification of two cases $g = AU(d_1, d_2)$, then g is the description of all that is common to (or shared by) d_1 and d_2. Therefore, anti-unification builds a symbolic similarity that describes *all* aspects in which two ore more cases are similar.

In the rest of the article we will use the formalism of feature terms to describe both generalizations and cases. A *feature term* was defined as follows:

Feature Terms Given 1) a signature $\Sigma = \langle S, F, \leq \rangle$ (where S is a set of sort symbols; F a set of feature symbols, and $\leq$ is a decidable partial order on S such that $\perp$ is the least element called any) and 2) a set υ of variables, a *feature term* is an expression of the form:

$$\psi ::= X : s[f_1 = \phi_1, ..., f_n = \phi_n] \qquad (1)$$

where X (the *root* of the feature term) is a variable in υ, s is a sort in S, $f_1, ..., f_n$ are features in F, $n \geq 0$, and each ϕ_i is in turn, a set of feature terms and variables.

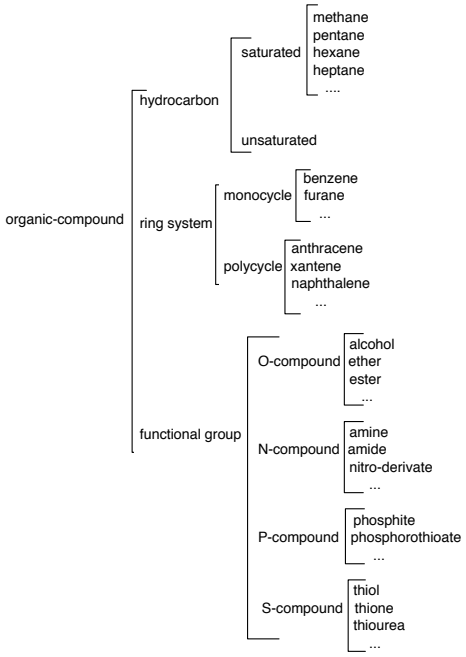


Figure 1: A sort hierarchy for the Toxicology ontology.

Notice that when $n = 0$ we are defining a variable with no features.

The partial order $\leq$ gives an informational order among sorts since $s_1 \leq s_2$ means that s_1 is a super-sort of s_2, i.e. s_2 is more specific than s_1. Notice that this order is the converse of $\geq_g$: $\leq$ is the more-specific-than relation while $\geq_g$ is the more-general-than relation. Using the partial order $\leq$ we can also define the *least upper bound (lub)* of two sorts as the most specific super-sort common to both sorts.

In order to illustrate *feature terms* and the notion of *lub* we will use examples of the Toxicology domain (ntp.niehs.nih.gov/ntpweb/). The goal is to classify a given chemical compound as positive or negative for carcinogenicity on both sexes of two rodent species: rats and mice. We used feature terms to describe the molecular structure of chemical compounds (Armengol & Plaza 2005) and also we defined an ontology based on the IUPAC nomenclature for chemical compounds. Figure 1 shows the sort hierarchy representing this chemical ontology. The most general sort is *organic-compound* and most specific sorts are the leafs of this hierarchy (e.g. *pentane*, *hexane*, *benzene*, *furane*, etc). Thus, when comparing two sorts, for instance *benzene* and *furane*, *organic-compound* is a super-sort of both. The least upper bound (the most specific super-sort) of *benzene* and *furane* is the sort *monocycle*. Similarly, the lub of *benzene* and *xantene* is the sort *ring-system*; and the lub of *methane* and *O-compound* is *organic-compound*.

Figure 2 shows an example of feature term. The *C-127* is a feature term of sort *organic-compound* with three features main-group, radical-set and p-radicals. The values of these features are in turn, feature terms with their par-

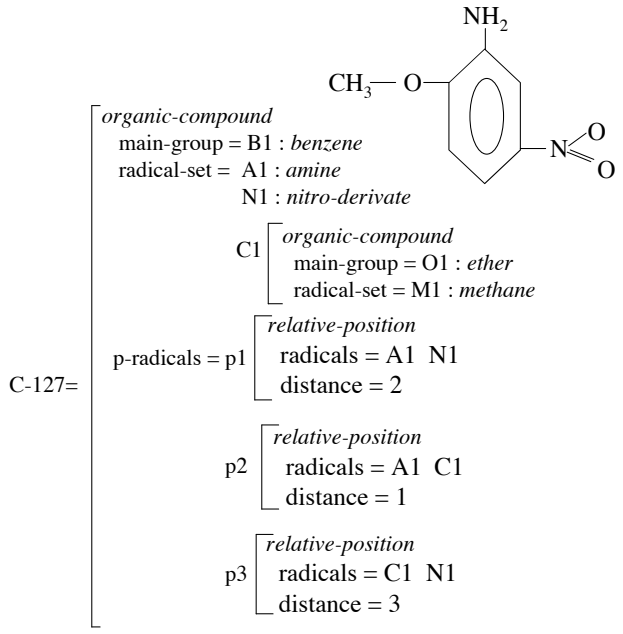


Figure 2: Description of the organic compound C-127, the *5-nitro-O-anisole*.

ticular features. Thus, the value of the main-group feature is a feature term B1 of sort *benzene* with no features. The feature radical-set has as value a set of three feature terms A1, N1 and C1 of sorts *amine, nitro-derivate* and *organic-compound* respectively. C1 has as features main-group and radical-set. The value of main-group is the feature term O1 of sort *ether*; and the value of radical-set is a feature term M1 of sort *methane*. The feature p-radicals of the feature term *C-127* is a set of three feature terms: p1, p2 and p3. The feature term p1 (as p2 and p3) of sort *relative-position* is described with two features: radicals and distance. The values of radicals are the same feature terms A1 and N1 in the feature radical-set. The value of distance is the number 2 (meaning that there is a distance of two carbon atoms between the radicals A1 and N1). The description of the feature terms p2 and p3 is similar to the p1 description.

Figure 3 shows the algorithm used to build the anti-unification of two feature terms. Given two feature terms D_1 and D_2 of sort s_1 and s_2 respectively, the algorithm creates a new feature term D of sort s. There are three possible cases: 1) when both D_1 and D_2 are of the same sort s, then D will be also of the sort s; 2) numbers are generalized to the sort *number*; and 3) when D_1 and D_2 have different sorts, the sort of D is the least general sort of both D_1 and D_2. The next step is to define the features of D. The features of D will be the features common to D_1 and D_2 and the value that each common feature f takes in D will be the anti-unification of the feature terms that are values of f in D_1 and D_2. When the value of the feature f is a set in at least one of the feature terms, then the *set-antiunification* function has to be applied.

```
Function AU (D1, D2)
  s1 := sort(D1)
  s2 := sort(D2)
  case
  1) s1 = s2 : return a ft D of sort s = s1
  2) s1 = number and s2 = number : return a ft D of sort s = number
  3) otherwise return a ft D of sort s = lub(s1, s2)
  end case
common := common-features (D1, D2)
for each f in common do
      v1 = D1.f
      v2 = D2.f
      if (v1 is a set) or (v2 is a set) then
         D.f := set-antiunification (v1, v2)
         D.f := AU (v1, v2)
      end if
end for
```

Figure 3: The anti-unification algorithm. $D.f$ stands for the value that D holds in the feature f.

Let the sets V_1 and V_2 be the values of f in D_1 and D_2 respectively. The anti-unification of V_1 and V_2 has to produce as result a set S of cardinality $min\{card(V_1), card(V_2)\}$. Each element in S is the anti-unification of an element (feature term) in V_1 and an element of V_2. Specifically, $AU(V_1, V_2)$ finds a set of values S such that:

- $card(S) = min\{card(V_1), card(V_2)\}$
- each $s_i \in S$ is the anti-unification of two values $v_j \in V_1$ and $v_k \in V_2$. The AU algorithm is applied to each possible pair (v_j, v_k) obtaining a set G with $V_1 \times V_2$ feature terms
- $AU(V_1, V_2)$ is the set of the most specific $card(S)$ elements in G. The selection of these elements has to be made taking into account that there are incompatible pairs of values. For instance, let us suppose that the most specific feature term in G has been obtained from the anti-unification of the pair (v_i, v_h) ($v_i \in V_1$ and $v_h \in V_2$). In such situation, all the $g_p \in G$ obtained from the anti-unification of v_i or v_h have to be rejected since they are incompatible with (v_i, v_h).

To illustrate this algorithm we will show the anti-unification of the chemical compounds *C-127* (Fig. 2) and *C-084* (Fig. 4). *AU(C-127, C-084)* is a feature term of sort *organic-compound*. The set of features common to *C-127* and *C-084* is {main-group, radical-set, p-radicals}. For each one of these common features, their values will be anti-unified recursively. Since the value of the feature main-group is *benzene* in both *C-127* and *C-084*, the value of main-group in *AU(C-127, C-084)* is also *benzene*.

The value of the feature radical-set is the set $V_1 = \{A1, N1, C1\}$ in *C-127* and the set $V_2 = \{A2, A3, C2\}$ in *C-084*. Thus, $AU(V_1, V_2)$ will be a set containing the three most specific feature terms. Table 1 shows the possible anti-unifications of all combinations of pairs of values (one from V_1 and the other from V_2). The column labeled as $AU(v_i, v_h)$ shows the anti-unification of the sorts of the

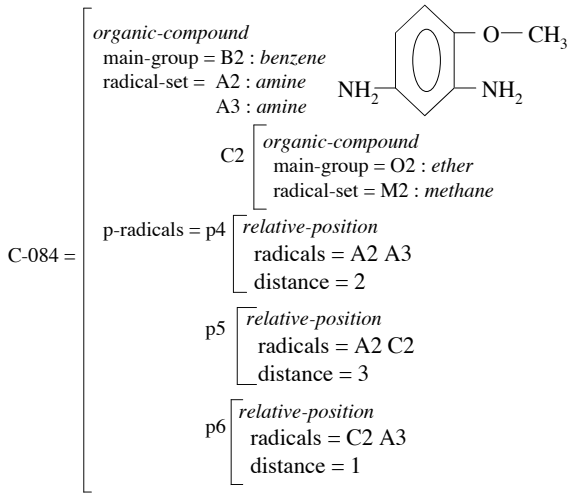


Figure 4: The chemical compound *C-084*, the *2,4 - diamino anisole*.

$AU(v_i, v_h)$	$v_i \in V_1$	$v_h \in V_2$
g_1 : *amine*	$A1$: *amine*	$A2$: *amine*
g_2 : *amine*	$A1$: *amine*	$A3$: *amine*
g_3 : *organic-comp*	$A1$: *amine*	$C2$: *organic-com*
g_4 : *N-compound*	$N1$: *nitro-derivate*	$A2$: *amine*
g_5 : *N-compound*	$N1$: *nitro-derivate*	$A3$: *amine*
g_6 : *organic-comp*	$N1$: *nitro-derivate*	$C2$: *organic-com*
g_7 : *organic-comp*	$C1$: *organic-comp*	$A2$: *amine*
g_8 : *organic-comp*	$C1$: *organic-comp*	$A3$: *amine*
g_9 : *organic-comp* main-group = O2 : *ether* radical-set = M2 : *methane*	$C1$: *organic-comp*	$C2$: *organic-com*

Table 1: Set G_{rs} containing all possible pairs of values of the feature radical-set, V_1 and V_2, and their anti-unification (AU).

two values of the pair (v_i, v_h). The set $AU(V_1, V_2)$ has to contain the three most specific $g_p \in G_{\mathsf{rs}}$.

The most specific feature term obtained from the pairs $(A1, A2)$ and $(A1, A3)$ is of sort *amine*. Notice that both pairs are incompatible, since both use $A1 \in V_1$. Thus, the first element of $AU(V_1, V_2)$ is a feature term of sort *amine* with no features. Let us assume that $g_1 = AU(A1, A2)$ is included in $AU(V_1, V_2)$. This means that pairs g_2, g_3, g_4 and g_7 are incompatible with g_1 and, therefore, they cannot be included in the set $AU(V_1, V_2)$. The next most specific feature term is $g_5 = AU(N1, A3)$, a feature term of sort *N-compound* without features. Finally, the next most specific feature term is g_9, a feature term of sort *organic-compound* that is the anti-unification of the feature terms $C1$ and $C2$ with the features main-group and radical-set. Therefore, $AU(V_1, V_2) = \{g_1, g_5, g_9\}$.

The feature p-radicals has as value the set $V_3 = \{p_1, p_2, p_3\}$ in *C-127* and the set $V_2 = \{p_4, p_5, p_6\}$ in *C-084*. Table 2 shows the set G_{pr} of all the possible combi-

$AU(v_i, v_h)$	$v_i \in V_1$	$v_h \in V_2$
g'_1 : *relative-position* radicals = *amine, N-compound* distance = 2	p_1	p_4
g'_2 : *relative-position* radicals = *amine, organic-comp* distance = number	p_1	p_5
g'_3 : *relative-position* radicals = *amine, organic-comp* distance = number	p_1	p_6
g'_4 : *relative-position* radicals = *amine, organic-comp* distance = number	p_2	p_4
g'_5 : *relative-position* radicals = *amine, organic-comp* distance = number	p_2	p_5
g'_6 : *relative-position* radicals = *amine, organic-comp* distance = 1	p_2	p_6
g'_7 : *relative-position* radicals = *organic-compound organic-compound* distance = number	p_3	p_4
g'_8 : *relative-position* radicals = *organic-compound N-compound* distance = 3	p_3	p_5
g'_9 : *relative-position* radicals = *organic-compound N-compound* distance = number	p_3	p_6

Table 2: Set G_{pr} containing all the possible combinations among the values of the feature p-radicals, V_3 and V_4, and their anti-unification (AU).

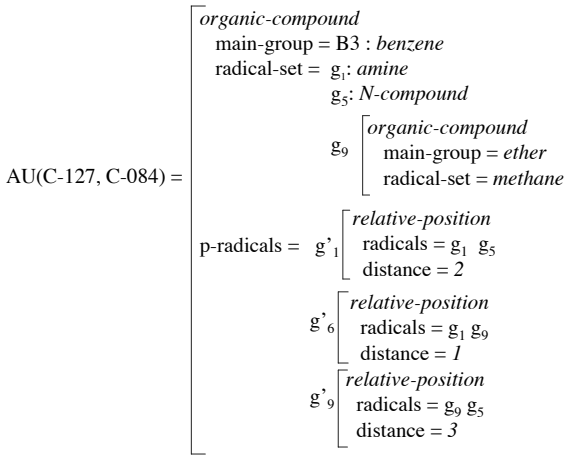


Figure 5: Anti-unification feature term of the chemical compounds *C-127* and *C-084*.

nations of the values in V_3 and V_4 and their anti-unfications. The column labeled as $AU(v_i, v_k)$ shows the complete feature term anti-unification of the values. The most specific feature terms are g'_1, g'_6 and g'_9 (they are not incompatible among them since they are the result of the anti-unification of different values from V_3 and V_4). Therefore, $AU(V_3, V_4) = \{g'_1, g'_6, g'_9\}$. Figure 5 shows the complete anti-unification feature term of the compounds *C-127* and *C-084*.

We have shown the anti-unification of two feature terms for simplicity sake, but anti-unification can be applied to a set of feature terms obtaining a new feature term with what is shared by all the cases of the set.

Using anti-unification for explanation

This section presents the way in which descriptions resulting from the anti-unification of a set of cases can be used to provide explanation of the classification of a new problem in CBR systems. Let CB be a case base containing cases classified in one of the solution classes $S = \{S_1, ..., S_m\}$. Let us suppose that p is a new problem to be solved and $C = \{c_1, ..., c_k\}$ the set of the k cases more similar to c. There are two possible situations:

- all the cases in C are in one class S_i
- the cases in C are in several classes

Concerning the first situation, most of CBR methods classify p as belonging to S_i and give as explanation of this classification the k cases in C. Our approach is that the explanation of why p is in S_i is given by what c shares with all the retrieved cases. In other words, the anti-unification $AU(c_1...c_k, p)$ is an explanation of why the cases in C have been considered as the more similar to p, since it is a description of all that is shared among the retrieved cases and the new problem. As an example, consider Fig. 6 where the problem is the chemical compound *C-068* and the *k*-NN algorithm with $k = 3$ retrieves as most similar the compounds

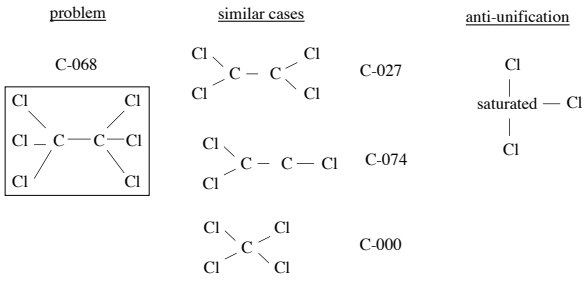


Figure 6: The $k = 3$ compounds that the *k-NN* algorithm retrieves as the most similar to *C-068*. The right part shows the anti-unification of the three most similar cases.

C-074, C-027, C-028. Since the three retrieved cases are carcinogenic for mice, *C-068* will also be classified as carcinogenic for mice. The explanation of this classification (right part of Fig. 6) is that all the compounds are saturated hydrocarbons with (at least) three chlorine (Cl) radicals.

However, very often the second situation above with multiple possible solution classes occurs. For simplicity we will our approach considering that some cases in C belong to one solution class (say S^+) and some others belong to another class (say S^-), but our explanation scheme is also applicable to situations with more than two classes.

Let $C^+ \subseteq C$ the subset of cases in class S^+, and $C^- \subseteq C$ the subset of cases in class S^- ($C = C^+ \cup C^-$). In addition to the particular classification of p by using the majority rule or some other aggregation criterion, the user should understand why the cases in C have been considered similar to p. As we justified in the first situation above, the anti-unification is a good explanation when all the cases in P belong to the same solution class but this is not the situation now. The explanation scheme we propose for this situation is composed of three descriptions:

- AU^*: the anti-unification of p with all the cases in C. This description shows what aspects of the problem are shared by all the retrieved cases, i.e. the k retrieved cases are similar to p because they have in common what is described in AU^*.
- AU^+: the anti-unification of p with the cases in C^+. This description shows what has p in common with the cases in C^+.
- AU^-: the anti-unification of p with the cases in C^-. This description shows what has p in common with the cases in C^-.

This explanation scheme supports the user in the understanding of the classification of a problem p. Figure 7 shows the intuitive idea of our approach. The problem p is on the border of the two solution classes. This means that it is similar both to some cases belonging to S^+ and to some other cases belonging to S^-. In fact, in the situation shown in figure 7 (p similar to 4 cases of the class S^+ and to 3 cases of the class S^-) the only reason to classify p in S^+ is that there is only one more case in C^+ than in C^-. With the

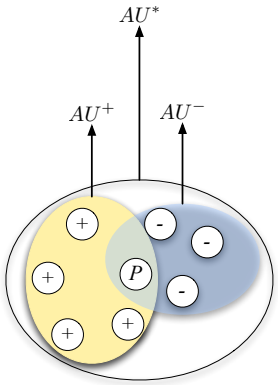
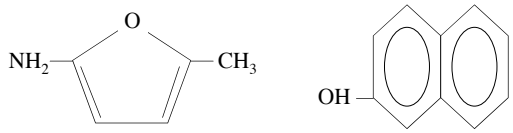


Figure 7: The sets of retrieved cases used to build the three anti-unification terms used in the explanation schema.

explanation scheme we propose, the similarities among p and the cases of each class are explicitly given to the user, who can decide the final classification of p. Thus, A^* is the anti-unification of all the cases considered as the most similar to p, i.e. is a description containing all the commonalities of the similar cases. When this description is too general (e.g. most of the features hold the most general sort as value), the meaning is that the cases have low similarity. Conversely, when A^* is a description with some features holding some specific value, this means that the cases share something more than only the general structure. For instance, the AU^* of the chemical compounds *6- hydroxynaphtalene* and the *2- amino, 3- methylfuran* (shown in Fig. 8) is a feature term that describes a molecule that is a ring system (since the *2- amino, 3- methylfuran* is a monocycle and the *6- hydroxynaphtalene* is a polycycle) holding one radical with no specific sort, since the *lub* of the alcohol (OH) and both the amine (NH_2) and the methane (CH_3) is *organic-compound*. Therefore, for this example the AU^* is not very informative. Instead, the explanation of the classification of the chemical compound *C-068* (Fig 6) gives more information since explains that all the compounds are saturated hydrocarbons with three chlorine radicals.



$$\text{AU}^* = \begin{bmatrix} \textit{organic-compound} \\ \text{main-group} = \textit{ring-system} \\ \text{radical-set} = \textit{organic-compound} \end{bmatrix}$$

Figure 8: Molecular structure of the *2- amino, 3- methylfuran* (left) and *6- hydroxynaphtalene* (right), and their anti-unification.

The AU^+ shows the commonalities among the problem p and the retrieved cases belonging to C^+. This allows the user to focus on those aspects that could be relevant to classify p as belonging to C^+. As before, the more specific is AU^+ the more information gives for classifying p. Notice that AU^+ could be as general as AU^*; in fact, it is possible that both feature terms are equal. This situation means that p has not too many similar aspects with the cases of C^+ that differ from those p shares with C^-. A similar situation may occur with AU^-.

Let us to illustrate the complete explanation scheme with an example on the Toxicology domain. In our example the goal is to assess the carcinogenicity of the chemical compound *C-356* for male rats shown in Fig. 9, that also shows the set C formed by five chemical compounds that have been assessed as the most similar cases to C-356. The set C can be partitioned in two subsets, namely C^+ containing those compounds that are *positive* for carcinogenesis, and C^- containing those compounds that are *negative* for carcinogenesis; specifically, $C^- = \{$*C-242, C-171*$\}$ and $C^+ = \{$*C-084, C-127, C-142*$\}$.

Following our approach, the explanation scheme for chemical compound *C-356* is as follows:

- The description AU^* is the chemical structure shown in the left of Figure 10; i.e. the compounds in C and *C-356* have in common that they are all benzenes with at least three radicals: one of these radicals is a functional group derived from the oxygen (i.e. an alcohol, an ether or an acid) called *O-compound* in the figure; another radical (called *rad1* in the figure) is in the position next to the functional group (chemically this means that both radicals are in disposition $ortho$). Finally, there is a third radical (called *rad2* in the figure) that is in no specific position.
- The description AU^- is the chemical structure shown in Figure 10, and shows that *C-356* and the chemical compounds in C^- have in common that they are benzenes with three radicals: one radical derived from an oxygen (*O-compound*), a radical *rad1* with another radical (*rad3* in the figure) in position *ortho* with the *O-compound*, and finally a third radical (*rad2*) with no specific position.
- The description AU^+ is the chemical structure in Figure 10, and shows that *C-356* and the chemical compounds in C^+ have in common that they are benzenes with three radicals: one of the radicals is derived from an oxygen (*O-compound*), another radical is an *amine* (NH_2) in position *ortho* with the *O-compound*, and the third radical (*rad1*) is at distance 3 of the *O-compound* (chemically this means that both radicals are in disposition $para$).

Using the majority rule, the compound *C-356* will be classified in the class C^+ (positive carcinogenesis) because $card(C^+) = 3$ and $card(C^-) = 2$. The explanation scheme shows to the user the feature term AU^* that states that all the retrieved compounds are benzenes with three radicals, one of them an *O-compound* in *ortho* position with respect another radical. Also, the feature term AU^- states that all the compounds with negative carcinogenesis (those in C^-) are also benzenes with three radicals. One of the

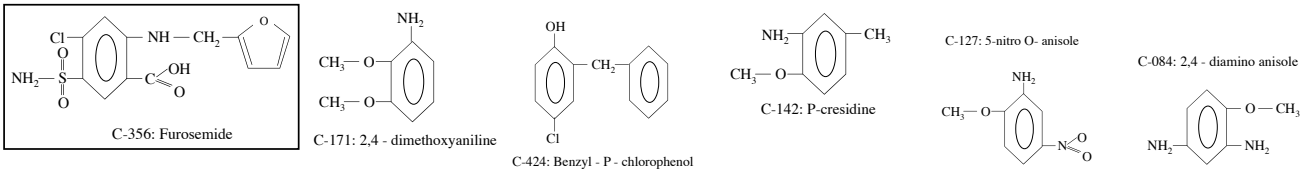


Figure 9: Molecular structure of the chemical compound *C-356* and the five compounds that have been retrieved as the most similar to *C-356*.

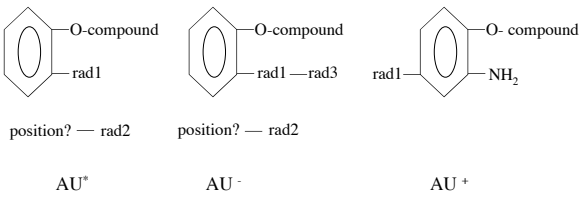


Figure 10: AU^* is the chemical structure common to all the compounds in Fig. 9. AU^- is the chemical structure common to C-356 and the negative compounds (i.e. *C-242* and *C-171*). AU^+ is the chemical structure common to *C-356* and the positive compounds (i.e. *C-084, C-127* and *C-142*).

radicals is an *O-compound* in position *ortho* with another radical that has, in turn, a radical. From both feature terms AU^- and AU^* the user may infer that the position *ortho* among the radical *O-compound* and the radical *rad1* is only important when *rad1* has radicals since in such situation all the compounds in C^- are negative for carcinogenesis.

On the other hand, as states the feature term AU^+, the compounds in C^+ are also benzenes with three radicals. One of these radicals is an *O-compound* that is in position *ortho* with a radical *amine* (NH_2) and in position *para* with another radical. By comparing both terms AU^+ and AU^- the user may conclude that both the kind of radical in position *ortho* with the *O-compound* and the position of the third radical are important to classify a compound as positive. In other words, from the descriptions AU^- and AU^+ the user is able to observe that the presence of the *amine* may hypothetically be a key factor in the classification of a compound as positive for carcinogenesis. Once the symbolic similarity description gives a key factor (such as the amine in our example), the user can proceed to search the available literature for any empirical confirmation of this hypothesis. In this particular example, a cursory search in the Internet shown that there is empirical evidence supporting the hypothesis of *amine* presence in aromatic groups (such as benzene) being correlated with carcinogenicity (Sorensen 2001; Ambs & Neumann 1996).

Finally, in situations where more than two classes are present in the retrieval set, our explanation scheme is simply to build one anti-unification description for each one of them. For instance, if cases in the retrieve set belong 4 classes the explanation scheme consists on the following symbolic descriptions: AU^*, AU^1, AU^2, AU^3, and AU^4.

Discussion

The anti-unification is the least general generalization of a set of cases. This means that it is described using the same features than the cases and that it is a description completely understood by the user. A very common explanation of the result of a CBR system is to give the set of cases C more similar to the problem p. The main shortcoming of this kind of explanation is that when the cases have a complex structure or when the solution has needed of some adaptation, the user can have some difficulties in understanding the solution of the problem at hand (Doyle, Tsymbal, & Cunningham 2003; McSherry 2004). Instead, using the explanation scheme we propose, the anti-unification AU^* gives a global justification of why the cases in C has been considered as the most similar and also gives a different description (i.e. AU^+ and AU^-) to justify the similarity of the problem p to each class, i.e. at first sight the user can understand why the problem could be classified as belonging to a class. Notice that this explanation scheme supports the user in taking the final decision to classify a problem when the cases more similar to p belong to different classes, but this explanation is independent on the classification produced by the CBR system.

The anti-unification is a generalization but is not a discriminant generalization for a class, i.e. it can cover not only the examples used to generalize but also some unseen examples of a different class. As Fig. 11 shows, the AU^- is the generalization of the problem p and the cases in C^-, nevertheless it can also cover some cases with positive carcinogenicity. The reason the anti-unification is not discriminant is that AU^- is built without using counter-examples, i.e. no case in C^+ is used.

Therefore, the anti-unification gives only an explanation for the problem at hand focusing on what is shared and describing all that is shared. However, it is not a discriminant description that distinguishes cases in C^+ from cases in C^-. For this purpose counterexamples should be used to obtain a generalization, say G^+ for C^+, such that G^+ covers every case in C^+ and none in C^-. Notice that $G^+ \geq_g AU^+$, and therefore does not contain all that is common to p and C^+. In fact, using a standard top-down induction technique to build G^+ we would usually obtain the *smallest* discriminant generalization. Although this approach is useful for inductive learning, from our point of view a lot of useful information about what is shared is lost. This is the reason to

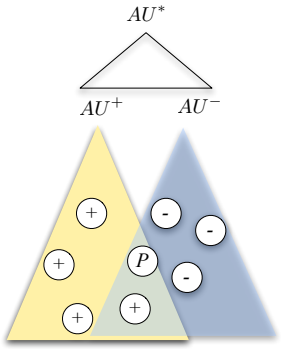


Figure 11: This example shows that the generalizations built by anti-unification are not discriminant descriptions.

use anti-unification for explanations instead of discriminant generalizations as those that can be built, foe example, using a decision tree induction technique.

Related Work

A common form of explanation in CBR is to show the user the case that has been considered as the most similar to the problem at hand. Nevertheless, there is a lot of work focusing on the appropriateness of this explanation (Cunningham, Doyle, & Loughrey 2003; McSherry 2004). Cunningham et al. performed some experiments on classification tasks in order to evaluate the importance of giving an explanation on the user acceptance of the result. They compared the acceptance of the results of two systems: a CBR and a rule-based system. The experiment showed that the results of the CBR with explanations were more convincing than those of the rule-based system.

McSherry (McSherry 2004) argues that the most similar case (in addition to the features that have been taken as relevant for selecting that case) also has features that could act as arguments against that case. For this reason, McSherry proposes that the explanation of a CBR system has to explicitly distinguish between the case features is favor of an outcome and the case features against it. In this way, the user could decide about the final solution of the problem. A similar idea is that proposed by (McCarthy *et al.* 2004) that use the differences among cases to support the user in understanding why some cases do not satisfy some requirements.

Our approach is based on generate an explanation scheme from the similarities among a problem and a set of cases. As the approaches of McSherry and McCarthy et al., the explanation scheme of our approach is also directed to the user. We make two assumptions: 1) a set C of the most similar cases has been generated from a CBR method, and 2) the cases in C can belong to different classes. From this set of cases, the explanation scheme shows the symbolic similarity of the problem with all the cases retrieved (AU^*) and also with the retrieved cases of each class ($AU^1, AU^2, \ldots, AU^k$). This means that the user can analyze the similarities and, by comparing the descriptions AU^*, $AU^1, AU^2, \ldots, AU^k$, can determine by herself the importance of the similarities and the differences among the descriptions. The difference of our approach with that of McSherry is that we explain the result using a set of similar cases whereas McSherry explains it using the similarities and differences within the most similar case compared to the problem at hand.

Other approaches, such that of Leake (Leake 1994) and Cassens (Cassens 2004), consider that the form of the explanation should be different depending on the user goals. These approaches are based on the idea that the explanation of the result cannot be based taking into account only one case or a small set of cases. Leake (Leake 1994) see the process of explanation construction as a form of goal-driven learning where the goals are those facts that need to be explained and the process to achieve them gives the explanation as result. Cassens (Cassens 2004) uses the Activity Theory to systematically analyze the evolution of an user in using a system, i.e. how the user model is changing. The idea is that in using a system the user can change his expectations about it and, in consequence, the explanation of the results would also have to change. In our approach we are considering classification tasks, therefore the user goals are always the same: to classify a new problem. This means that the explanation has to be convincing enough to justify the classification and we assume that the kind of explanation has always the same form, i.e. it does not change along the time.

In this paper we used the notion of symbolic similarity to produce explanations on the performance of CBR systems. In addition, to show the retrieved cases to the user, our proposal also shows the most specific generalizations covering the retrieved cases and the new problem.

Since CBR systems perform lazy learning, and lazy learning builds local approximations of the target concepts, we can view the explanations in this framework. For instance, the retrieved cases in C^+ are an *extensional description* of the local approximation to the carcinogenicity concept, while the most specific generalization AU^+ is an *intensional description* of the local approximation to the carcinogenicity concept. Thus, our approach complements the classical explanation in CBR based on extensional descriptions of the local approximation with several intensional descriptions (AU^*, AU^+, and AU^-) that allow the user to focus on what is shared (and not shared) among the new problem and the retrieved cases.

The idea of *symbolic similarity* was introduced in (Armengol & Plaza 2003) but was there used to build a discriminant generalization. In this approach, a *symbolic similarity* description is considered as a local approximation of a class description. This local approximation is obtained using the most relevant features of the new problem; then cases that do not satisfy this approximation were discarded. The result is a symbolic description that is satisfied only by cases that belong to one of the classes; thus, that description can be considered as a partial description of the class. This symbolic description can then be used to explain why a problem has been classified into a class and the cases covered by that description form the retrieved set that can be shown also to the user as endorsing the system prediction.

Conclusions and Future Work

In this paper we focused on the problem of how to explain the user the classification given by a CBR system. In particular, we assumed that the outcome of the CBR is a set of the k cases considered as the most similar (under some specific criteria) to the problem at hand. These k cases can belong to different classes, therefore the system has to explain to the user both a) why these cases have been considered as the most similar to the problem (even they belong to different classes), and b) why the problem could be classified in each one of these classes. Our approach allows to give an explanation scheme composed by several general descriptions, each one explaining different aspects of the outcome. Thus, one of the descriptions (A^*) shows the features shared by the problem and all the retrieved cases; therefore, the user can understand why these k cases have been considered as the most similar. Each one of the other descriptions ($AU^1, AU^2, \ldots, AU^k$) shows the similarities among the new problem and the subset of cases belonging to each class. These descriptions show the user the features that the problem shares with the cases of a class. We saw that using this explanation scheme the user can easily understand why these cases have been retrieved and also (as in the example we described) can detect which parts of these descriptions are relevant to discriminate among the classes. In addition, the analysis of the explanation scheme can support the user in doing an oriented search in the literature.

There are several lines of research spawning from the approach presented here that we plan to pursue. Concerning the toxicology domain, current ML and statistical techniques have shown limited a proficiency in prediction (Helma & Kramer 2003); the explanation scheme using symbolic similarity that we provide seem to be helpful in improving our understanding of this domain.

Another line of future research is the use of the symbolic similarity descriptions in a CBR system for purposes of self-assessment. We are interested in developing confidence measures that could allow a CBR system to reliably assess its confidence in each specific prediction. The symbolic similarity descriptions we use will cover in general positive and negative cases with respect to a solution class, and this fact can be used to estimate a degree of confidence in a predicted solution. There are several ways in which this assessment can be made and experiments in several data sets are needed to determine their usefulness.

Finally, symbolic similarity descriptions could be used to determine in an adaptive way the granularity of the local approximations; for instance, in a CBR system using *k*-nearest neighbor symbolic similarity descriptions could be used to determine for each specific problem which value of *k* offers a better confidence in the predicted solution.

Acknowledgements This work has been supported by the project SAMAP (TIC2002-04146-C05-01). The authors also thanks to Dr. Lluis Bonamusa for his assistance in developing the representation of the chemical compounds.

References

Ambs, S., and Neumann, H. G. 1996. Acute and chronic toxicity of aromatic amines studied in the isolated perfused rat liver. *Toxicol. Applied Pharmacol.* 139:186–194.

Armengol, E., and Plaza, E. 2003. Remembering similitude terms in case-based reasoning. In *3rd Int. Conf. on Machine Learning and Data Mining MLDM-03*, number 2734 in Lecture Notes in Artificial Intelligence, 121–130. Springer-Verlag.

Armengol, E., and Plaza, E. 2005. An ontological approach to represent molecular structure information. In J. L. Oliveira et al., ed., *Biological and Medical Data analysis, ISBMDA05*, Lecture Notes in Computer Science, –. Springer.

Cassens, J. 2004. Knowing what to explain and when. In *Proceedings of the ECCBR 2004 Workshops. Technical Report 142-04*, 97–104. Departamento de Sistemas Informáticos y Programación, Universidad Complutense de Madrid, Madrid, Spain.

Cunningham, P.; Doyle, D.; and Loughrey, J. 2003. An evaluation of the usefulness of case-based explanation. In *Proceedings of the* 5^{th} *International Conference on Case-based Reasoning (ICCBR 2003)*, 122–130. Springer.

Doyle, D.; Tsymbal, A.; and Cunningham, P. 2003. A review of explanation and explanation in case-based reasoning. In *Technical report TCD-CS-2003-41*. Department of computer Science. Trinity college, Dublin.

Helma, C., and Kramer, S. 2003. A survey of the predictive toxicology challenge 2000-2001. *Bioinformatics* 1179–1200.

Leake, D. 1994. Issues in goal-driven explanation. *Proceedings of the AAAI Spring symposium on goal-driven learning* 72–79.

McCarthy, K.; Reilly, J.; McGinty, L.; and Smyth, B. 2004. Thinking positively - explanatory feedback for conversational recommender systems. In *Proceedings of the ECCBR 2004 Workshops. Technical Report 142-04*, 115–124. Departamento de Sistemas Informáticos y Programación, Universidad Complutense de Madrid, Madrid, Spain.

McSherry, D. 2004. Explanation in recommendation systems. In *Proceedings of the ECCBR 2004 Workshops. Technical Report 142-04*, 125–134. Departamento de Sistemas Informáticos y Programación, Universidad Complutense de Madrid, Madrid, Spain.

Plaza, E.; Armengol, E.; and Ontañón, S. 2005. The explanatory power of symbolic similarity in case-based reasoning. *Artificial Intelligence Review. Special Issue on Explanation in Case-based Reasoning* to appear.

Sorensen, R. U. 2001. Allergenicity and toxicity of amines in foods. In *Proceedings of the IFT 2001 Annual Meeting, New Orleans, Louisiana*.