Multiple-Instance Case-Based Learning for Predictive Toxicology

Eva Armengol and Enric Plaza

IIIA – Artificial Intelligence Research Institute,
CSIC – Spanish Council for Scientific Research,
Campus UAB, 08193 Bellaterra, Catalonia, Spain
{eva,enric}@iiia.csic.es

Abstract. Predictive toxicology is the task of building models capable of determining, with a certain degree of accuracy, the toxicity of chemical compounds. Machine Learning (ML) in general, and lazy learning techniques in particular, have been applied to the task of predictive toxicology. ML approaches differ in which kind of chemistry knowledge they use but all rely on some specific representation of chemical compounds. In this paper we deal with one specific issue of molecule representation, the multiplicity of descriptions that can be ascribed to a particular compound. We present a new approach to lazy learning, based on the notion of *multiple-instance*, which is capable of seamlessly working with multiple descriptions. Experimental analysis of this approach is presented using the Predictive Toxicology Challenge data set.

1 Introduction

There are thousands of new chemicals registered every year around the world. Although these new chemicals are widely analyzed before their commercialization, the long-term effects of many of them on the human health are unknown. The National Toxicology Program (NTP) started with the goal of establish standardized bioassays for identifying carcinogenic substances (see more information at http://ntp-server.niehs.nih.gov). These bioassays are highly expensive in time and money since they take several years and sometimes their results are not conclusive. The use of automatic tools could support the reduction of these costs. In particular, artificial intelligence techniques such as knowledge discovery and machine learning seem to be specially useful.

The goal of Predictive Toxicology is to build models that can be used to determine the toxicity of chemical compounds. These models have to contain rules able to predict the toxicity of a compound according to both the structure and the physical-chemical properties. A Predictive Toxicology Challenge (PTC) [15] was held in 2001 focusing on machine learning techniques for predicting the toxicity of compounds. The toxicology data set provided by the NTP contains descriptions of the bioassays done on around 500 chemical compounds and their results on rodents (rats and mice) of both sexes.

There are two open problems in predictive toxicology: 1) representing the chemical compounds, and 2) determining which characteristics of chemical compounds could be useful for classifying them as toxic or not toxic (i.e. the toxicity model). A summary of both the different representations and the methods used to build the toxicity model proposed in the PTC can be found in [4]. Basically, there are two families of representations: those based on *structure-activity relationship (SAR)* and those based on the compound substructures. SAR are equation sets that relate molecular features and that allow the prediction of some molecular properties before the experimentation in the laboratory. Approaches based on compound substructures (*relational representation*) represent a chemical compound as a set of predicates relating the atoms composing the molecule. Most authors, independently of the kind of compound representation, use inductive learning methods to build a toxicity model.

In [3] we introduced a new relational representation based on the chemical nomenclature and also a lazy learning technique to assess the toxicity of compounds. The main difference between our approach and those of the PTC is that we do not try to build a toxicity model, but we assess specifically the toxicity of each new chemical compound. This is because lazy learning techniques are problem-centered, i.e. they solve a new problem based on its similarity to other problems previously solved. In the toxicology domain, lazy learning techniques assess the toxicity of a chemical compound based on its similarity to other chemical compounds with known toxicity.

In particular, in [3] we proposed to use the k-NN algorithm [10] for assessing the toxicity of a chemical compound. Because chemical compounds are represented using feature terms [2] (i.e. they are structured objects) we defined a new similarity measure called Shaud to be used in the k-NN algorithm. Results obtained with the lazy learning approach using the feature terms representation of the compounds are comparable to the results obtained using inductive approaches. Moreover, in our representation only the molecular structure is taken into account whereas SAR approaches use a lot of information related with properties of the molecules and also results of some short-term assays.

Since our representation of molecules is based on chemical nomenclature, and this has some ambiguity issues we propose to use the notion of *multiple-instance* [11] in lazy learning techniques. Specifically, the ambiguities in chemistry nomenclature stem from the fact that often a single molecule can be described in several ways, i.e. it may have synonymous names. The notion of multiple-instance precisely captures the idea that an example for a ML technique can have multiple descriptions that, nonetheless, refer to the same physical object. Therefore, this paper proposes two new techniques for integrating multiple-instances into k-NN methods and performs their experimental evaluation in the toxicology domain.

This paper is organized as follows: first we describe the issues involved in representing chemical compounds; then Section 2 presents Shaud, a similarity measure for structured cases, and the new multiple-instance techniques for k-NN; an empirical evaluation is reported in section 4, and finally a conclusions section closes the paper.
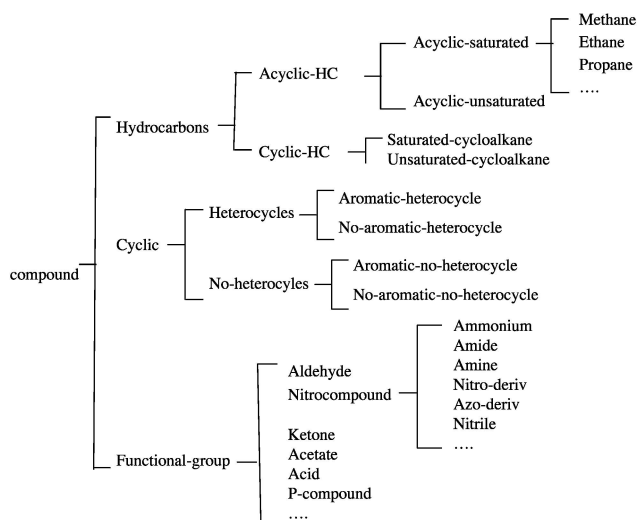
Fig. 1. Partial view of the chemical ontology

2   Representation of the Chemical Compounds

We propose using a representation of chemical compounds based on the *chemical ontology* used by experts in chemistry. We represent compounds as a structure with substructures using the chemical ontology that is implicit in the nomenclature of the compounds. Fig. 1 shows part of the chemical ontology we have used to represent the compounds in the Toxicology data set. This ontology is based on the IUPAC chemical nomenclature which, in turn, is a systematic way of describing molecules. In fact, the name of a molecule provides all the information needed to graphically represent the structure of the molecule.

According to the chemical nomenclature rules, the name of a compound is formed in the following manner: *radicals' names + main group*. The *main group* is often the part of the molecule that is either the largest or the part located in a central position. However, there is no general rule for forming the compound name. *Radicals* are groups that are usually smaller than the main group. A main group can contain several radicals and a radical can, in turn, have a new set of radicals. Both main group and radicals are the same kind of molecules, i.e. the benzene may be the main group in one compound and a radical in some others.

In our representation (see Fig. 2) a chemical compound is represented by a feature term of sort *compound* described by two features: main-group and p-radicals. The values of the feature main-group belong to some of the sorts shown in Fig. 1. The value of the feature p-radicals is a set whose elements are of sort *position-radical*. The sort *position-radical* is described using two features: radicals and position. The value of radicals is of sort *compound*, as the whole chemical compound, since it has the same kind of structure (a main group with radicals). The feature position indicates where the radical is bound to the main group.
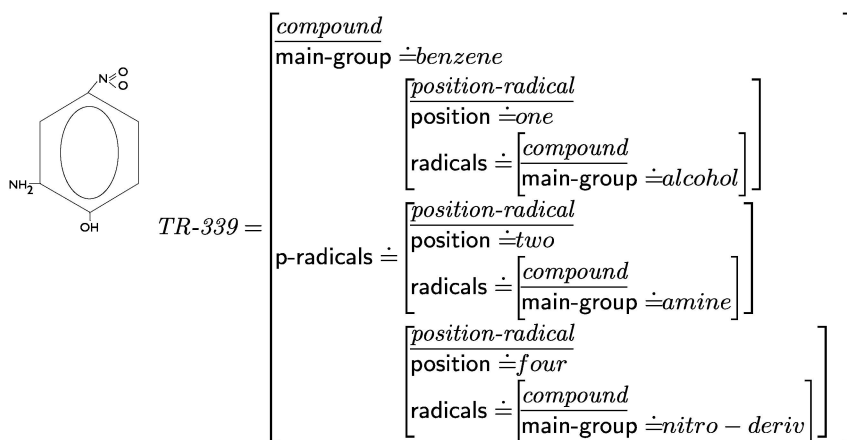
$$TR\text{-}339 \doteq \begin{bmatrix} compound \\ \text{main-group} \doteq benzene \\ \text{p-radicals} \doteq \begin{bmatrix} \begin{bmatrix} position\text{-}radical \\ \text{position} \doteq one \\ \text{radicals} \doteq \begin{bmatrix} compound \\ \text{main-group} \doteq alcohol \end{bmatrix} \end{bmatrix} \\ \begin{bmatrix} position\text{-}radical \\ \text{position} \doteq two \\ \text{radicals} \doteq \begin{bmatrix} compound \\ \text{main-group} \doteq amine \end{bmatrix} \end{bmatrix} \\ \begin{bmatrix} position\text{-}radical \\ \text{position} \doteq four \\ \text{radicals} \doteq \begin{bmatrix} compound \\ \text{main-group} \doteq nitro-deriv \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Fig. 2. Representation of TR-339, *2-amino-4-nitrophenol*, with feature terms
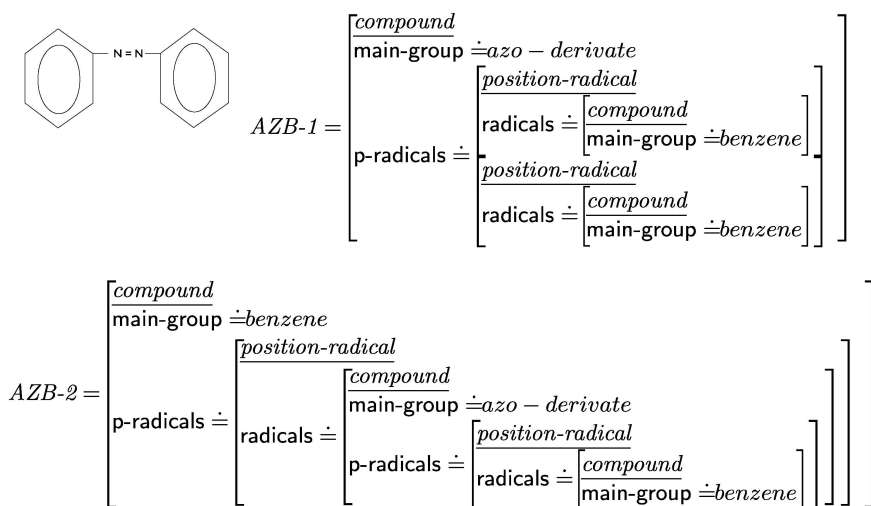
For example, the chemical compound TR-339, *2-amino-4-nitrophenol* (Fig. 2), has a benzene[1] as main group and a set of three radicals: an *alcohol* in position one; an *amine* in position two; and a *nitro-derivate* in position four. Note that this information has been directly extracted from the chemical name of the compound following the nomenclature rules. Moreover, this kind of representation is very close to the representation that an expert has of a molecule from the chemical name.

Nevertheless, the chemical nomenclature is ambiguous. For instance, from the name *2-amino-4-nitrophenol*, chemists assume that the main group of the molecule is the benzene and that the radicals are in positions 1, 2 and 4. In this molecule the name is clear because the benzene is the largest group and chemists have a complete agreement in considering the main group. Nevertheless, the name of some other molecules is not so unambiguous. For instance, the chemical compound TR-154 of the toxicology database is the *azobenzene* (Fig. 3) a compound with a benzene as main group. This compound is also known as *diphenyldiimide* where the main group is an *azo-derivate* (structurally equivalent to a *diimide*). Therefore, we say that *azobenzene* and *diphenyldiimide* are *synonyms*.

Due to these ambiguities, we propose to take into account synonyms regarding the structure of the molecule. Thus, the *2-amino-4-nitrophenol* has several possible synonyms taking into account different positions of the radicals (although they are not strictly correct from the point of view of the chemical nomenclature): we could consider that the amine is in position 1, the alcohol in position 2 and the nitro-derivate in position 5. Notice that the difference between the synonymous representations is the position of the radicals.

Dietterich *et al.* [11] introduced the notion of *multiple-instance*. This notion appears when a domain object can be represented in several alternative ways.

[1] The *phenol* is a benzene with an alcohol as radical in position one.

$$AZB\text{-}1 = \begin{bmatrix} compound \\ \text{main-group} \doteq azo - derivate \\ \text{p-radicals} \doteq \begin{bmatrix} position\text{-}radical \\ \text{radicals} \doteq \begin{bmatrix} compound \\ \text{main-group} \doteq benzene \end{bmatrix} \\ position\text{-}radical \\ \text{radicals} \doteq \begin{bmatrix} compound \\ \text{main-group} \doteq benzene \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$AZB\text{-}2 = \begin{bmatrix} compound \\ \text{main-group} \doteq benzene \\ \text{p-radicals} \doteq \begin{bmatrix} position\text{-}radical \\ \text{radicals} \doteq \begin{bmatrix} compound \\ \text{main-group} \doteq azo - derivate \\ \text{p-radicals} \doteq \begin{bmatrix} position\text{-}radical \\ \text{radicals} \doteq \begin{bmatrix} compound \\ \text{main-group} \doteq benzene \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Fig. 3. Graphical represetation of the molecular structure of *azobenzene* and two synonymous descriptions (AZB-1 and AZB-2) of *azobenzene*

This situation is very common in domains such as chemistry where a molecule can be seen from several points of view. In particular, when addressing the problem of determining whether a molecule is active. Multiple instances are needed because a molecule can have several conformations some of which can be active and some others not. We propose to use the notion of multiple-instance to represent the compounds of the toxicology data set. We represented 360 compounds of the PTC data set using feature terms. When a compound can have several synonymous representations we defined a feature term for each alternative representation, i.e. there are multiple instances for the compound. Fig. 3 shows the synonymous representations using feature terms of the *azobenzene*: one of them considers the *benzene* as the main group and the other considers the *azo-derivate* as the main group.

Thus, for each one of the 360 chemical compounds of the data set we defined as many instances as necessary to capture the different synonyms of a compound according to its structure. For some compounds, the differences between synonyms are the positions of the radicals since in all them we considered the same main group. Instead, some other compounds have synonyms with different main group. This is the case of the *azobenzene* in Fig. 3 where AZB-1 has an *azo-derivate* as main group and AZB-2 has a *benzene* as main group. As it will be explained later, although a compound to be classified is compared with all the synonymous descriptions of each compound, the final classification takes into account only the similarity with one of the synonyms. In other words, for classification purposes the data set contains 360 chemical compounds even most of them have several synonymous representations.

In the next section we explain how k-NN algorithm can be modified in order to deal with the synonymous representations of the compounds.
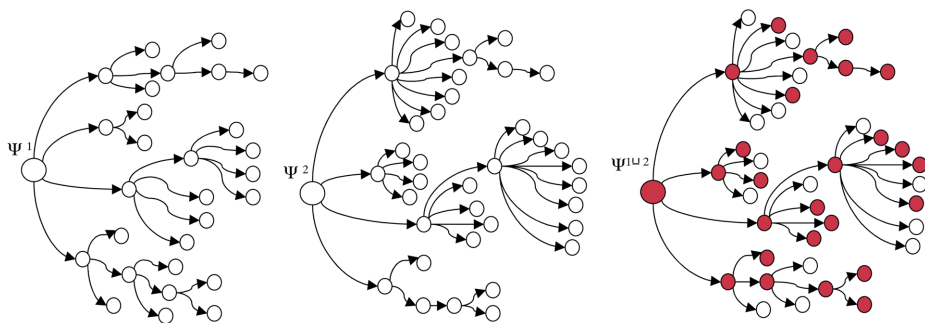
Fig. 4. ψ^1 and ψ^2 are feature terms represented as graphs. $\psi^{1 \sqcup 2}$ is a feature term containing both the *shared structure* (shaded nodes) and the *unshared structure* (white nodes) of ψ^1 and ψ^2

3   Similarity of Relational Cases

In order to assess the toxicity of a chemical compound we proposed the use of lazy learning techniques. In particular, we use the *k nearest neighbor (k-NN)* [10] algorithm. Given a new problem p and a case-base B containing solved problems, the k-NN retrieves from B the k cases that are most similar to p. There are several similarity assessments to be used in the k-NN algorithm [21] but all of them work on objects represented as a set of feature value pairs. Nevertheless, we represent the chemical compounds as feature terms, i.e. they have a structured representation and we proposed Shaud [3] as a similarity measure for relational cases represented as feature terms. The main idea of Shaud is to assess the similarity between two feature terms taking into account their structure. When comparing the structure of two feature terms ψ^1 and ψ^2 (see Fig. 4), there are two parts that have to be taken into account: 1) the part of the structure that is common to both ψ^1 and ψ^2, called the *shared structure* (shown by shaded nodes in Fig. 4); and 2) the part of the structure that is present in ψ^1 but not in ψ^2 and vice versa, called the *unshared structure* (shown by white nodes in Fig. 4). Shaud assesses the similarity of two feature terms ψ^1 and ψ^2 by computing the similarity of the shared structure and then normalizing this similarity value taking into account both the shared and the unshared structure.

Let us suppose that the k most similar cases to the new problem p belong to several classes. In such a situation, a common criteria for assessing a solution class to p is the *majority criterion*, i.e. p is classified as belonging to the solution class that most of the k of the retrieved cases belong to. We experimented with Shaud using the majority criterion but results were not satisfactory enough since the accuracy in classifying non-toxic compounds was clearly higher than the accuracy in classifying toxic ones. For this reason, we proposed a new classification criterion for k-NN called *Class Similarity Average* (CSA). CSA is not domain-dependent and in [3] we proved that it improves the accuracy on both toxic and non-toxic compounds.
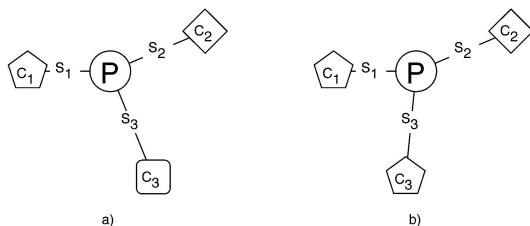
Fig. 5. Two situations of 3-nearest neighbor with similarity values s_1, s_2 and s_3: a) three different cases are retrieved, and on b) two of the cases, c_1 and c_3 are synonymous (since they have the same shape)

For each compound p to be classified as toxic or non-toxic, Shaud yields the similarity between p and each one of the k most similar cases. Then CSA computes the average of the similarity of the cases in the same class; then the class with higher average similarity is selected as the solution for p. More formally, let the *positive* class be the set of chemical compounds that are toxic (or carcinogenic) and the *negative* class the set of chemical compounds that are non-toxic. Let A^+ be the *positive retrieval set*, i.e. the set containing the retrieved cases belonging to the positive class, and A^- be the *negative retrieval set*, i.e. the set containing the retrieved cases belonging to the negative class. The carcinogenic activity of a compound p is obtained according to the CSA criterion, where the average similarity for both retrieval sets is computed as follows:

$$sim^+ = \frac{1}{|A^+|}\sum_{c_i \in A^+} s_i \text{ and } sim^- = \frac{1}{|A^-|}\sum_{c_i \in A^-} s_i$$

and then the compound p is assigned to one of the classes according to the *decision rule*:

> if $sim^+ < sim^-$ then p belongs to the positive class
> else p belongs to the negative class

3.1   Lazy Learning Techniques with Multiple-Instances

The CSA criterion assumes that the k most similar cases are different chemical compounds. Nevertheless, this assumption is not true when using multiple-instances since some of the retrieved cases can be, in fact, different representations of the same compound. For instance, Fig. 5.a represents a situation where P is the new problem to classify and $k = 3$. Cases c_1 and c_2 and c_3 are the three cases most similar to P with similarities s_1, s_2 and s_3 respectively. c_1 is the most similar to P and c_3 is the least similar. Let us assume that c_1 and c_3 belong to the positive class and c_2 belongs to the negative class. The classification of P can be done using the CSA criterion and the decision rule as explained above. Fig. 5.b shows a situation where c_1 and c_3 are synonymous (they have the same shape in the figure). Therefore, for $k = 3$ we have two cases (since two of them are synonyms); clearly, we cannot treat this situation as identical to that of 5.a.

Notice that, since c_1 and c_3 are synonyms we have two similarity values (s_1 and s_3). How can we now decide whether P is positive or negative?

Let us now consider the synonymy relation ($\cong$) among the set of retrieved cases $A = A^+ \cup A^-$. Assume, for instance, that A^+ (or equivalently A^-) has a pair of synonymous cases $c \cong c'$. We can build a reduced retrieval set $\bar{A}^+$ without synonyms simply by selecting one of the synonymous and discarding the other; i.e. we could take as the reduced retrieval set either $\bar{A}^+ = A^+ \setminus \{c\}$ or $\bar{A}^+ = A^+ \setminus \{c'\}$. Now we introduce two techniques, Shaud-MI_{max} and Shaud-MI_{av}, to deal with multiple-instances using reduced retrieval sets $\bar{A}^+$ and $\bar{A}^-$.

The technique Shaud-MI_{max} selects the synonymous case in the retrieval set A^+ (resp. A^-) with greatest similarity value and discards the others. For instance, if $c \cong c'$ and they have similarity values s and s' respectively, if $s > s'$ then c is selected and thus the reduced retrieval set is $A^+ \setminus \{c'\}$. Let us call the synonymous case $\bar{c}$ with maximal similarity value the *canonical representative of a collection of synonyms* $c_1 \cong c_2 \cong \ldots \cong c_m$ and let $\bar{s} = max(s_1, s_2, \ldots, s_m)$ be its similarity. Clearly, if a case c has no synonyms in A^+ then $\bar{c} = c$ and $\bar{s} = s$. We will define the reduced retrieval set $\bar{A}^+$ as the collection of canonical cases of A^+. The same process is used for obtaining $\bar{A}^-$ from A^-. Finally, the solution class is computed by modifying the CSA criterion as follows:

$$sim^+ = \frac{1}{|\bar{A}^+|} \sum_{\bar{c}_i \in \bar{A}^+} \bar{s}_i \text{ and } sim^- = \frac{1}{|\bar{A}^-|} \sum_{\bar{c}_i \in \bar{A}^-} \bar{s}_i \qquad (1)$$

and the same CSA decision rule ($sim^+ < sim^-$) is used as before.

For instance, in the situation shown in Fig. 5.b if the synonyms c_1 and c_3 belong to the positive class, then $\bar{A}^+ = \{c_1\}$, $|\bar{A}^+| = 1$, and $\bar{s}_1 = max(s_1, s_3) = s_1$. Analogously, if c_2 belongs to the negative class we will have that $\bar{s}_2 = s_2$ and, following the CSA decision rule, P will be classified as positive since $s_1 > s_2$ and thus $sim^+ > sim^-$.

The technique Shaud-MI_{av} is similar to the previous one except that it uses an average criterion instead of the maximum criterion. Thus, for any collection of synonyms $c_1 \cong c_2 \cong \ldots \cong c_m$ in a retrieval set their average similarity $\bar{s} = \frac{1}{m}(s_1 + s_2 + \ldots + s_m)$ is computed. Let the canonical synonymous case $\bar{c}$ be a randomly chosen case from a set of synonymous cases $c_1 \cong c_2 \cong \ldots \cong c_m$. As before, if c has no synonyms on A^+ then $\bar{c} = c$ and $\bar{s} = s$. Let $\bar{A}^+$ be the reduced retrieval set with the canonical cases of A^+, and for each $\bar{c}_i \in \bar{A}^+$ let $\bar{s}_i$ be the average synonymous similarity computed as indicated above, then the CSA average similarity is again computed as in expression (1) with the same decision rule as before.

For instance, in the situation show in Fig. 5.b if the synonymous c_1 and c_3 belong to the positive class, then $\bar{A}^+ = \{c_1\}$ (i.e. $|\bar{A}^+| = 1$), and $\bar{s}_1 = \frac{s_1 + s_3}{2}$. Following the CSA decision rule, P will be classified as positive when $sim^+ > sim^-$ and negative otherwise.

Table 1. Distribution of the NTP compounds on the four data sets

data set	Positive	Negative	Equivocal	Inadequate	Unknown
MR	127	176	39	6	12
FR	101	205	35	7	12
MM	102	195	37	13	13
FM	124	198	19	7	12

Table 2. Accuracy results in the four toxicology data sets for Shaud similarity with three aggregation criteria CSA, MI_{max}, and MI_{av}

Shaud	k	MR Acc	TP	FP	FR Acc	TP	FP	MM Acc	TP	FP	FM Acc	TP	FP
CSA	3	54.43	.522	.431	61.77	.463	.319	58.47	.428	.329	56.16	.438	.368
	5	54.66	.560	.466	58.63	.520	.373	58.83	.491	.353	**57.97**	.512	.377
MI_{max}	3	58.37	.517	.362	**64.86**	.461	.257	**59.42**	.403	.315	57.21	.445	.346
	5	**59.28**	.515	.343	64.54	.498	.285	57.62	.443	.352	56.34	.496	.394
MI_{av}	3	57.39	.505	.363	64.73	.458	.256	59.26	.439	.302	57.25	.474	.362
	5	58.15	.549	.355	63.85	.466	.274	56.05	.423	.372	56.47	.483	.383

4 Experiments

In our experiments we used the toxicology data set provided by the NTP. This data set contains around 500 chemical compounds that may be carcinogenic for both sexes of two rodents species: rats and mice. The carcinogenic activity of the compounds has proved to be different for both species and also for both sexes. Therefore, there are in fact four data sets.

We solve the predictive toxicology problem as a classification problem, i.e. for each data set we try to classify the compounds as belonging to either the *positive* class (carcinogenic compounds) or to the *negative* class (non-carcinogenic compounds). We used 360 compounds of the data set (those organic compounds whose structure is available in the NTP reports) distributed in the classes as shown in Table 1.

The experiments have been performed with the k-NN algorithm using Shaud as distance and taking Shaud-MI_{av} and Shaud-MI_{max} explained in the previous section. Results have been obtained by the mean of seven 10-fold cross-validation trials. Table 2 shows these results in terms of accuracy and true positives (TP) and false positives (FP) for both options and also we compare them with the version of CSA without multiple-instances. Concerning the accuracy, the versions with multi-instances taking $k = 3$ improve the version without multi-instances, especially in MR and FR data sets. Nevertheless, taking $k = 5$, the versions with multi-instances are better on rats (i.e. MR and FR) but the accuracy does not improves on mice (i.e MM and FM). We are currently analyzing why the prediction task on mice is more difficult than in rats.

Currently machine learning methods are evaluated using ROC curves [13]. A ROC curve is a plot of points (FP, TP) where TP is the ratio between positive

cases correctly classified and the total number of positive cases; and FP is the ratio between negative cases incorrectly classified and the total number of negative cases. The line $x = y$ represents the strategy of randomly guessing the class and the point $(0, 1)$ represents perfect classification. Points above the diagonal are preferred since they represent a higher number of TP than FP. Thus, a point is better than another if TP is higher and FP is lower. Moreover, given two points (FP_1, TP_1) and (FP_2, TP_2) such that $FP_1 < FP_2$ and $TP_1 < TP_2$ the performance of the two methods is incomparable and the cost of false positives has to be taken into account in order to choose between them. The convex hull of a set of points is the smallest convex set that includes the points. Provost and Fawcett [18] introduced the notion of convex hull in the ROC curves as a way to compare machine learning methods. They prove that (FP, TP) points on the convex hull correspond to optimal methods whereas those points under the convex hull can be omitted since they never reach an optimal performance.

We will use the ROC convex hull to compare Shaud-CSA, Shaud-MI_{max} and Shaud-MI_{av} to the best methods of the PTC. According to the final conclusions of the PTC ([20]) best methods for each data set are the following:

- MR. Gonzalez [14]
- FR. Kwansei [17], Viniti [6]
- MM. Baurin [5], Viniti, Leuven [7]
- FM. Viniti, Smuc (from [20])

Figures 6 and 7 show the ROC points of the methods above for all the data sets. We included in these figures the points corresponding to the Shaud versions.

4.1   Discussion

Concerning the MR data set, the methods of Kwansei (3), Gonzalez (2) and Viniti (6) are in the convex hull of the PTC, so they are the best methods for this data set (if we do not take into account the cost). Shaud-CSA (7 and 8), Shaud-MI_{max} (9 and 10) and Shaud-MI_{av} (11 and 12) are above the convex hull (for both $k = 3$ and $k = 5$). Taking the points separately we see that Shaud-MI_{max} (10) and Shaud-MI_{av} (12) both with $k = 5$ clearly improve the performance of Viniti and Smuc (5) method in the central zone. From our point of view, Shaud-MI_{max} is incomparable with Gonzalez and Kwansei methods since it increases the number of TP but also increases the number of FP. Therefore, choosing between these methods will depend on the cost assigned to the FP.

The best methods of the PTC for the FR data set are Viniti (6) and Kwansei (3). With respect to the convex hull, our methods do not perform very well but looking separately at the points we consider that Viniti and Kwansei are incomparable since Viniti produces few FP but also few TP. Instead, Kwansei clearly produces more TP. Choosing between these two methods depends on the cost of the FP and also on the necessity to detect as many TP as possible. In this sense, our methods are close to Kwansei. In particular Shaud-MI_{av} with both $k = 3$ and $k = 5$ is the best approach.

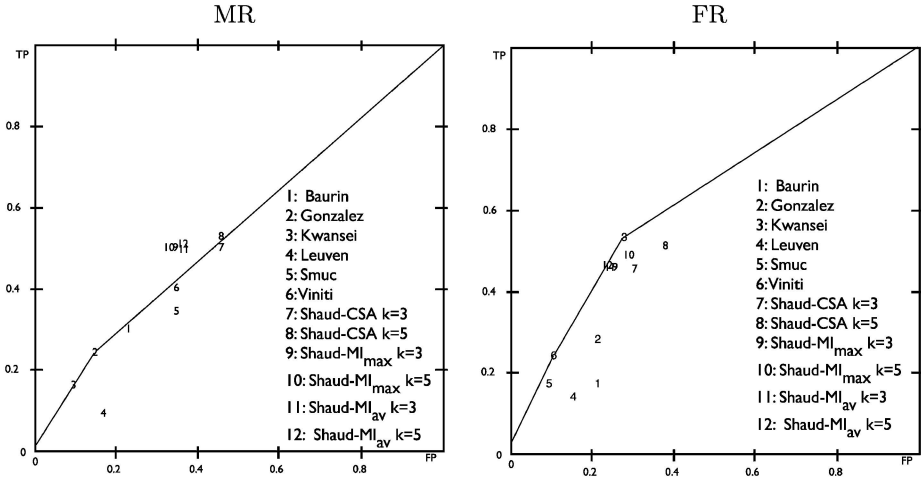**Fig. 6.** ROC curves of 12 methods for MR and FR data sets
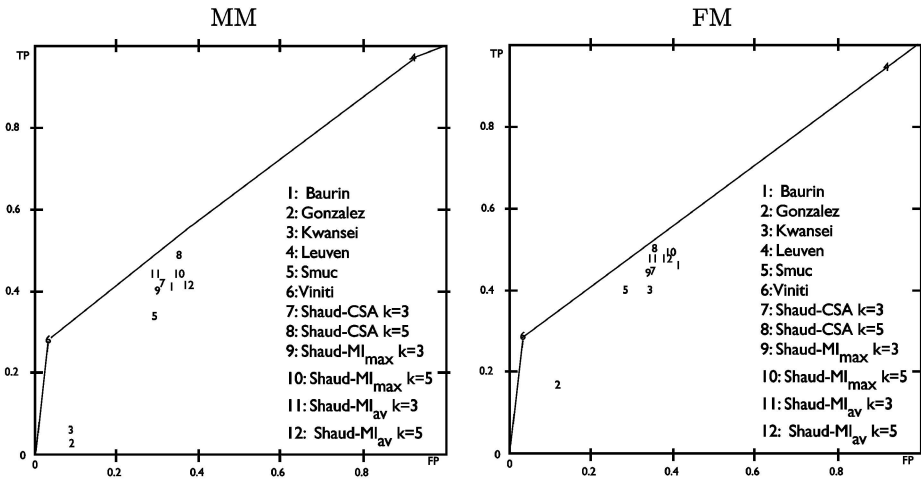

Fig. 7. ROC curves of 12 methods for MM and FM data sets

Concerning the MM data set, the best methods of the PTC are Viniti (6) and Leuven (4). The Viniti method is really excellent because the number of FP is low and the number of TP is high enough. Nevertheless, the Leuven method produces more TP although the number of FP is also very high. Our methods are in an intermediate position, near to the Baurin (1) method. In particular, any of the multiple-instances versions with any k has a number of TP near to that of Baurin but with fewer FP. All versions with $k = 3$ improve the Baurin method whereas CSA without multi-instances (8) and Shaud-MI_{max} (10) with $k = 5$ have higher TP but also higher FP. Shaud-MI_{av} has approximately the same number of TP but the higher number of FP.

Finally, concerning the FM data set, Viniti (6) and Leuven (4) methods are on the convex hull. Nevertheless we consider that the Leuven method is not so good since it is near to the (1,1) point. Our methods are near to Baurin (1), Kwansei (3) and Smuc (5). Smuc method is better than Kwansei since both have aproximately the same number of TP but Kwansei produces more FP. We consider that all our methods improve the Baurin method since the (FP, TP) points are on the left-hand side of Baurin (i.e the number of FP is lower) and all the versions with $k = 5$ produce more TP. The choice between any of our methods and Viniti or Smuc clearly depends on the cost of the FP.

Summarizing, establishing a cost measure is necessary in order to meaningfully choose the adequate methods for each data set. Nevertheless, our lazy approach using multiple-instances has proved to be competitive enough. A final remark is that most of the best methods use many information about the domain. Moreover methods based on the SAR representaton produce toxicity models in terms of molecular features that sometimes are not easy to determine. The Viniti method uses a domain representation that takes benefit ot the molecular structure, nevertheless this representation and also the toxicity model are difficult to understand. Instead, we used a representation close to the chemical nomenclature. In this representation we only taken into account the molecular structure without any additional feature. Our conclusion is that having only structural information is enough to obtain a comparable performance and it is not necessary to handle features that are neither intuitive nor easy to compute.

5   Related Work

The notion of *multiple-instances* is useful when domain objects can be viewed in several ways. Specifically, Dieterich et al. [11] used *multiple-instances* for determining the activity of a molecule, taking into account that a molecule has different isomers with different activity. As explained in section 2, chemical nomenclature allows synonym names for one compound. We intend to use the notion of *multiple-instances* to manage synonymous descriptions of compounds.

The basic idea of multiple-instances is that a domain object can be represented in several alternative ways. Chemistry is an application domain where multiple-instances can be applied in a natural way since the molecular structure of a compound has several possible configurations with different properties (e.g. a configuration may be active whereas another is inactive). Most of authors working on multiple-instances use chemical domains such *mutagenesis* [19] and *musk* (from the UCI repository). Dieterich et al. [11] introduced the notion of multiple-instance and they extended the *axis-parallel rectangle* method to deal with it. Other authors then proposed extensions of some known algorithms in order to deal with multiple-instances.

Chevaleyre and Zucker [8] proposed an extension of propositional rule learning. Specifically, they proposed two extensions of the RIPPER method [9]: NAIVE-RIPPERMI, that is a direct extension, and RIPPERMI which performs relational learning. Maron and Lozano-Perez [16] introduced a probabilistic mea-

sure, called *diverse density*, that computes the intersection of the *bags* (sets of synonymous objects) minus the union of the negative bags. Maximizing this measure they reach the goal concept. Zucker [22] introduced the *multi-part problem* meaning that an object can be represented by means of a set of descriptions of its parts. They propose extensions of the classical concept learning algorithm by defining a multiple entropy function and a multiple coverage function.

There are also some approaches using multiple-instances with a lazy learning approach. Wang and Zucker use the k-NN algorithm with the Hausdorf distance [12] defined to assess the distance between two bags. They introduce two versions of k-NN: *Bayessian k-NN*, which uses a Bayesian model to vote the final classification; and *citation k-NN* where the different bags are related in the same way as the references on information science.

6   Conclusions

In previous work we have shown that using both a chemical ontology based representation of the compounds and a lazy learning approach is a feasible approach for the predictive toxicology problem. However, our approach was limited by the fact that the ontology we were using (namely the chemical nomenclature) allows multiple descriptions of single compounds. Since the PTC data set we were using only used one description for each compound the selection of that description over the other ones introduced an unwanted and unknown bias. In fact, using Shaud similarity compared two compound descriptions but not the alternative descriptions that were not included in the data set; therefore results could be different if the selected descriptions were different.

Therefore, our purpose as explained in this paper was to use multiple descriptions when meaningful, but it was not enough to expand the PTC data set to allow every example to have several compound descriptions: we needed to define how multiple compound descriptions would be interpreted by lazy learning methods. In this paper we have introduced the notion of *reduced retrieval sets* to integrate Dietterich's notion of multiple-instances into k-NN techniques. Specifically, we consider that k-NN retrive k cases similar to a problem P and, for each class to whch P can be assigned, a retrieval set can be built from the retrieved cases of that class.

We presented two methods for dealing with multiple-instances, Shaud-MI_{max} and Shaud-MI_{av}, that specify how *reduced* retrieval sets are built from classical k-NN retrieval sets. This building process is, in fact, a specification of how to interpret the fact that more than one description of a specific compound are in the k-NN retrieval sets. Since Shaud-MI_{max} uses the maximal similarity among synonyms, the interpretation is that we only take into account the synonymous description that is the most similar disregarding the others. Nevertheless, multiple-instances are useful since they allow to find more similar matches in the k-NN retrieval process.

On the other hand, Shaud-MI_{av} uses the average similarity among retrieved synonyms, thus in some way penalizing multiple-instances that have a second

most similar compound description with a lower similarity value. Recall that both techniques normalize the aggregate similarities of the retrieval sets with the number of retrieved examples (i.e. not counting synonyms), and therefore a k-NN retrieval set contains k cases that represent different chemical compounds (as it would be without multiple-instances, which is exactly what Shaud-CSA does).

The experiments have shown that introducing multiple-instances improves the performance of lazy learning in general terms. Specifically, using multiple-instances improves results in the rats (both male and female) data sets, while in the mouse data sets using multiple-instances or not gives incomparable results (a cost measure would be needed to decide the best among them). Notice also that our lazy learning techniques are more competitive, when compared with other ML methods, in the rats data sets, while they are not distinguishable from other methods in the mouse data sets. Although the reasons for the differences in performance for lazy learning (and for the other ML methods) on the PTC is not well understood (see [15]) it seems that multiple-instances can be useful for the situations where a lazy learning method is adequate, as in the data set for male and female rats.

The representation of the chemical compounds in our experiments use only structural information. Instead, representations based on SAR use features for which computation is imprecise and which are not totally comprehensible by the expert. In the future, we plan to extend our representation to introduce information about short-term experiments. In particular, the Ames test [1] that has proved to be very important result and is easy to obtain.

Acknowledgements

References

1. B.N. Ames and J. McCann. Detection of carcinogens as mutagens in the salmonella/ microsome test: Assay of 300 chemicals: Discussion. In *Proceedings of the National Academy of Sciences USA*, volume 73, pages 950–954, 1976.
2. E. Armengol and E. Plaza. Bottom-up induction of feature terms. *Machine Learning*, 41(1):259–294, 2000.
3. E. Armengol and E. Plaza. Relational case-based reasoning for carcinogenic activity prediction. *Artificial Intelligence Review*, 20(1–2):121–141, 2003.
4. E. Armengol and E. Plaza. Lazy learning for predictive toxicology based on a chemical ontology. In W. Dubitzky and F.J. Azuaje, editors, *Artificial Intelligence Methods and Tools for Systems Biology*. In Press. Kluwer Academic Press, 2004.
5. N. Baurin, C. Marot, J.C. Mozziconacci, and L. Morin-Allory. Use of learning vector quantization and BCI fingerprints for the predictive toxicology challenge 2000-2001. In *Proceedings of the Predictive Toxicology Challenge Workshop, Freiburg, Germany*, 2001.

6. V. Blinova, D. Bobryinin, V. Finn, S. Kuznetsov, and E. Pankratova. Toxicology analysis by means of simple JSM method. *Bioinformatics*, 19(10):1201–1207, 2003.

7. H. Blockeel, K. Driessens, N. Jacobs, R. Kosala, S. Raeymaekers, J. Ramon, J. Struyf, W. Van Laer, and S. Verbaeten. First order models for the predictive toxicology challenge 2001. In *Proceedings of the Predictive Toxicology Challenge Workshop, Freiburg, Germany*, 2001.

8. Y. Chevaleyre and J.D. Zucker. Solving multiple-instance and multiple-part learning problems with decision trees and rule sets. application to the mutagenesis problem. Morgan Kauffman, 1995.

9. W. Cohen. Fast effective rule induction. In *Proceedings of the 12th International Conference on Machine Learning*, pages 204–214, 2001.

10. B. V. Dasarathy. *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques.* Washington; Brussels; Tokyo; IEEE Computer Society Press, 1990.

11. T. Dietterich, R. Lathrop, and T. Lozano-Perez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence Journal*, 89(1-2):31–71, 1997.

12. G.A. Edgar. *Measure, Topology, and Fractal Geometry.* Springer Verlag, 1995.

13. J. P. Egan. *Signal Detection Theory and ROC Analysis.* Series in Cognition and Perception. New York: Academic Press, 1975.

14. J. Gonzalez, L. Holder, and D. Cook. Application of graph-based concept learning to the predictive toxicology domain. In *Proceedings of the Predictive Toxicology Challenge Workshop, Freiburg, Germany*, 2001.

15. C. Helma and S. Kramer. A survey of the predictive toxicology challenge 2000-2001. *Bioinformatics*, 19(10):1179–1182, 2003.

16. O. Maron and T. Lozano-Perez. A framework for multiple instance learning. *Neural Information Processing Systems*, (10):–, 1998.

17. H. Owada, M. Koyama, and Y. Hoken. ILP-based rule induction for predicting carcinogenicity. In *Proceedings of the Predictive Toxicology Challenge Workshop, Freiburg, Germany*, 2001.

18. F. Provost and T. Fawcett. Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. In *Proceedings of the KDD-97*, 1997.

19. A. Srinivasan, S. Muggleton, r.D. King, and M.J.E. Sternberg. Mutagenesis: ILP experiments in a non-determinate biological domain. In *Proceedings of the Fourth Inductive Logic Programming Workshop*, 1994.

20. H. Toivonen, A. Srinivasan, R. King, S. Kramer, and C. Helma. Statistical evaluation of the predictive toxicology challenge. pages 1183–1193, 2003.

21. D. Wettschereck and T. G. Dietterich. Locally adaptive nearest neighbor algorithms. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems*, volume 6, pages 184–191. Morgan Kaufmann Publishers, Inc., 1994.

22. J. Zucker. A framework for learning rules from multiple instance data. In P. Langley, editor, *European Conference on Machine Learning*, pages 1119–1125, 2000.