ELSEVIER

Chemometrics and Intelligent Laboratory Systems 43 (1998) 135–145

Chemometrics and intelligent laboratory systems

Hybrid toxicology expert system: architecture and implementation of a multi-domain hybrid expert system for toxicology

Giuseppina Gini [a,*], Vito Testaguzza [a], Emilio Benfenati [b], Roberto Todeschini [c]

[a] *Dipartimento di Elettronica e Informazione, Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milan, Italy*
[b] *Laboratory of Environmental Chemistry and Toxicology, Department of Environmental Health Sciences, Istituto di Ricerche Farmacologiche 'Mario Negri', Milan, Italy*
[c] *Dipartimento di Scienze dell'Ambiente e del Territorio, Università Statale di Milano, Milan, Italy*

Received 9 January 1998; accepted 21 June 1998

Abstract

A hybrid expert system prototype using artificial neural networks (ANN) and classical rules has been developed for predicting toxicology of compounds. Modularity was a must for the architecture of the system. The study of chemicals was approached by establishing classes. When appropriate descriptors are calculated for the molecule, the ANN classifier assigns the chemical class to the compound. Then the toxic activity is quantitatively predicted of by one of the trained ANN in the system. After that, a qualitative prediction (active/non-active) is made by a rule-based system, calling only the correct knowledge base (KB) for the assigned class. This last step enabled us to give an explanation of the results. All the rules in the KBs have been obtained with automated learning techniques. © 1998 Elsevier Science B.V. All rights reserved.

Keywords: Toxicology; Expert systems; Artificial neural networks; Feature selection; QSAR models; WHIM descriptors; Automated rule extraction

1. Introduction

The increasing number of pollutants in the environment raises the problem of toxicological characterization of these chemicals. Toxicology is the science that defines the limits of safety of chemical agents [1].

The traditional way of assessing the toxic risk of a compound is to test them in animals. The results are then extended to humans using safety factors and dose relationships.

This approach, however, suffers many drawbacks [2]:

- cost of the experiments (> 1 million US$ per compound);
- the duration of the tests (3–5 years);
- public pressure to reduce or eliminate the use of animals in scientific experiments.

To overcome these problems, several computer-aided tools have been developed to help toxicologists assess the risk for new compounds.

Many algorithms have been proposed to explain toxic effects in different situations where homogeneous classes of chemicals showed various activities. However, in most cases these algorithms are suitable

* Corresponding author. Tel.: +39-2-2399-3626; Fax: +39-2-2399-3411; E-mail: gini@elet.polimi.it

0169-7439/98/$ - see front matter © 1998 Elsevier Science B.V. All rights reserved.
PII: S0169-7439(98)00125-7

for predictions only within the structure space spanned by the set of compounds used to build the model.

Attempts are now being made to model non-homogeneous sets of compounds, which is of course harder than explaining the properties of analogues [3]. This paper is a contribution to this area of study and to the application of computer science and artificial intelligence to real problems.

Several expert systems (ES) have been claimed to predict toxicity of chemicals. These ES currently use different approaches, based on a knowledge base (KB) of explicit rules derived from the knowledge of human experts, or relying on statistical approaches. The advantage of rule-based systems is that the set of rules (which are independent) can be extended without rebuilding the system. However, the disadvantages are that it is very hard to obtain a complete set of rules and that control knowledge is not easily integrated.

Intelligent technologies include various artificial intelligence techniques, namely ES, artificial neural networks (ANN), and rule induction. These fall along a continuum with subsymbolic processing at one end and symbolic processing at the other. Until recently, problem solvers typically used a single technique to build the solution. One way to deal with really complex systems is to combine two or more techniques in order to exploit their different strengths and overcome their weaknesses. The development of intelligent systems for practical applications can benefit from combinations of different techniques because no single technique can do everything.

There has been a considerable amount of research into integrating connectionist and symbolic processing. While this approach has clear advantages, it also involves serious difficulties and challenges. The hybrid approach is based on the use of two complementary paradigms, and aims at their synergistic combination in systems comprising both neural and symbolic components. Hybrid systems are becoming more common and useful. In fact the success of ANN may well reflect the ease with which it incorporates information processing approaches. However, it is still debated which engineering method to apply for developing effective hybrid systems.

In this project we aimed at several targets. First of all, studying state of the art systems will help clarify the needs of an ES in toxicology. An architecture fulfilling these needs can then be developed. One of the main tools we plan to use is ANN whose ability to model unknown and/or non-linear relationships is widely recognized. These abilities are now strongly supported on a theoretical basis. [1] We also intend to use inductive learning algorithms to extract the KB needed for the rule based ES.

The general target is then to integrate both these components into an architecture, and develop it in order to maximize the predictive power of the system.

2. Toxicology and advanced computer systems

In recent years, several ES have been proposed in toxicology, based on various approaches:

- rule-based systems drawn from human experts,
- systems using statistical methods,
- systems based on mechanistic processes.

Rule-based ES are suitable when some of the information is uncertain or even unknown. This is very common in toxicology. The technique a human expert usually uses to assess the potential hazard related to a compound is the subjective idea of *similarity* with other molecules. The expert looks for certain reactive groups that are known from the literature to have toxic action. Several reactive groups can be found in Ref. [4]. Examples of systems following this idea have been presented [5,6].

TOPKAT and *CASE* systems use the statistical approach. TOPKAT [7,8] is inspired by the classical QSAR principles. In any TOPKAT module we find a QSAR model and a database. CASE, developed by Klopman [9], Rosenkranz and Klopman [10–12], and Klopman and Rosenkranz [13,14] works statistically, comparing the fragments present in several training sets.

The only system using the mechanistic approach is *COMPACT* [15]. This depends very much on knowledge of the key biochemical processes in the activity of the compounds.

[1] Theorem by Hornik, Stinchombe and White (1989).

For a comprehensive overview of all these systems see Ref. [16].

3. The general architecture of the present system

All the programs presented in the literature (see above) lack descriptions of the computer science aspects. This is probably because most of these papers were published in toxicology or chemical journals. It is not clear how the inference engine of the rule based systems works. Moreover, most of these systems are not properly ES, since they do not show any symbolic reasoning. Additionally, the ability to explain the results is not confirmed.

Since uncertainty is a major characteristic of the toxicology field, it would be desirable to estimate the precision of the prediction. Statistical methods can do this easily, but rule-based systems cannot.

From the study of the existing systems, we established the features needed for an ES in toxicology.

- Ability to *explain* the results.
- Integration of the *quantitative computational approach with the classical rule based ES*. Two modules should be present. The quantitative module will predict the toxicity value, giving ample information, but no explanation can be given. The qualitative module, represented by a classical rule-based system, will predict the activity/non-activity of the compound, with limited information, but here we can show which rule explains the results.
- Choice of the *chemical class before studying toxic activity*; it is impossible to obtain good predictive models for heterogeneous sets of compounds. We, therefore, have to concentrate our modelling efforts on a limited number of classes. This demands an efficient classification module.
- *Estimate of the reliability of prediction.*
- *Modularity*: this is crucial since more classes could be added as modules.

A reductive hypothesis at the basis of our architecture is to approach the toxicology problem considering the *whole molecule* and not some of its fragments.

Fig. 1 illustrates the general architecture of the system. The inputs are the molecular descriptors, and the result is the toxicity value. The classification module assigns a chemical class to the compound, so

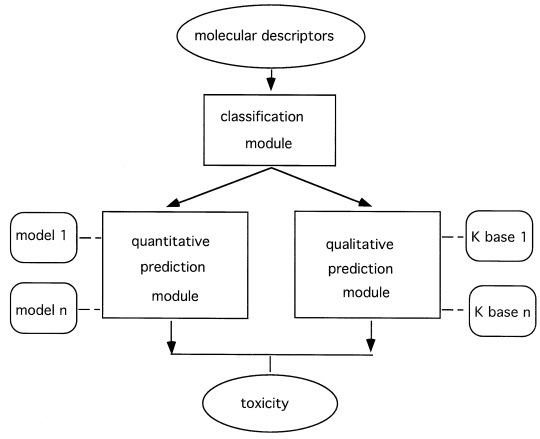

Fig. 1. General architecture.

that only the appropriate quantitative model and knowledge base will be activated. These two modules give the final result: a number, indicative of predicted toxicity or activity, a class (active/non-active), and an explanation for this conclusion.

As a result, we have a hybrid multi-domain ES, where the correct knowledge is activated only when necessary.

4. Molecular descriptors

Informative data representation is essential to obtain reliable results from ANN. Given the compound structure, that can be entered graphically, there are different ways to compute descriptors that account better for the geometry, physics and activity of the molecule. We had several choices: 'classical' such as physico-chemical, and topological.

There are different physico-chemical descriptors and parameters: log P, Henry coefficient, steric parameters, electronic parameters, etc. [17]. Their advantage is a firm physico-chemical basis (clear and straightforward interpretation). The main disadvantages are that some of them can be measured experimentally, though their availability and accuracy are low, and others can be calculated by several programs, but their value depends on the program used.

Topological indices [18] try to model the pattern of connections of atoms in a molecule. A great advantage is that they are available for all organic structures, and are easily computed. The disadvantages are many: difficult interpretation (especially for

high order connectivity indices), no direct chemical meaning, none of the topological variables or their principal components have any relationship to the spatial and conformational aspects of the compound, and finally, they generally do not encode information about 3-D aspects of molecules and have little information about the electronic characteristics of atoms.

The new tendency is to search a holistic approach to defining the compound. New descriptors are important in order to represent the structure of the molecule better. *WHIM* (weighted holistic invariant molecular) *descriptors* [19,20] are a recent approach to obtain a holistic view of the molecule. Their calculation starts from the molecular 3-D coordinates of the molecule in the lowest energy conformation (equilibrium). Their aim is to capture the relevant 3-D information regarding molecular size, shape symmetry and atom distribution with respect to some invariant reference frame. The algorithm involves a principal component analysis (PCA) on the centered molecular coordinates, using different weighting schemes. These descriptors have already proved their utility in modelling toxic properties of chemicals [21,22].

The intrinsic correlation among them can be removed by using variable subset selection (VSS) in regression analysis. The main advantage is their easy interpretation; each descriptor refers to an intuitive elementary geometric property of the molecule. This is an important characteristic for writing an easily explainable and usable KB.

5. Experimental

5.1. Data and descriptors

We chose to work on two data-sets: triazines and haloarenes. The toxic properties to predict were phytotoxicity [2] for triazines, pRB [3] and pAHH [4] for haloarenes. Both the data-sets were studied and supplied with 34 descriptors for each of the approximately 70 compounds per data-set at the Department of Environmental Sciences of Milan University. The molecular descriptors for these chemicals were the WHIM indicated above and calculated using the WHIM-3D/QSAR package, now available on request [23].

5.2. The classification module

To implement the classification module, we needed an algorithm that gave good results with soft decay of performances when questioned with unknown patterns. We used the LVQ NN classification algorithm introduced by Kohonen [24]; this program is available on Internet [5] and widely documented in Ref. [25]. LVQ is a supervised learning algorithm for statistical classification. Its purpose is to define *class regions* in the input data space.

Let $x \in \Re^n$ be the input vector to be classified. We take a finite number of 'codebook vectors' (free parameter vectors) and place them in the input space to approximate various domains of x by their quantified values. Usually, we assign several 'codebook vectors' for each class of x values.

Let $m_i \in \Re^n$, $i = 1,2,\ldots k$ be our 'codebook vectors'. The x is assigned to the class to which the nearest m_i belongs.

Let m_c be the nearest m_i to x:

$$m_c \text{ is chosen so that } \|x - m_c\| = \min_i\{\|x - m_i\|\}$$
$$\text{or } c = \operatorname{argmin}_i\{\|x - m_i\|\}.$$

Let $x(t)$ be a sample of input:

$$m_c(t+1) = m_c(t) + \alpha(t)\left[x(t) - m_c(t)\right]$$

if x and m_c belong to the same class,

$$m_c(t+1) = m_c(t) - \alpha(t)\left[x(t) - m_c(t)\right]$$

if x and m_c belong to different classes,

$$m_i(t+1) = m_i(t) \text{ for } i \neq c$$

for $0 < \alpha(t) < 1$, $\alpha(t)$ can be optimized for each $m_i(t)$ introducing $\alpha i(t)$.

[2] Phytotoxicity is defined as pI_{50}, that is the reciprocal of the molar concentration necessary to inhibit the reduction of the electronic acceptor by 50%.

[3] $p\text{RB} = -\log EC_{50}(\text{RB})$ where $EC_{50}(\text{RB})$ values are the in vitro rat hepatic cytosolic Ah RB (receptor binding) affinities.

[4] $p\text{AHH} = -\log EC_{50}(\text{AHH})$, where $EC_{50}(\text{AHH})$ are the in vitro induction of AHH (aryl hydrocarbon hydrolase).

[5] The LVQ_PAK is available at: cochlea.hut.fi. It is downloadable with anonymous ftp features.

The aim of these experiments was to correctly distinguish triazines and haloarenes. The training data-set comprised 110 samples (60 triazines and 50 haloarenes) and the validation set 37 samples (14 triazines and 23 haloarenes). The number of free parameter vectors was always 14. The number of iteration was set at $40 * 14 = 560$, where 40 is a number recommended by the authors of the algorithm. All possible combinations of the available options were tried.

5.3. The quantitative prediction module

The theorem by Hornik, Stinchombe and White (1989) ensures that a regressive feed-forward ANN is a universal function approximator from $\Re^n \rightarrow \Re$. This makes ANN a very powerful tool for mathematical modelling, especially for non-linear relationships. We used the back-propagation algorithm [26] on a three-layer neural model.

To avoid local minima we introduced a momentum term. To accelerate the convergence of the algorithm, we used the techniques explained in Ref. [27]. The code used in the experiments was made available by Davide Anguita [6] of the Department of Biophysical and Electronic Engineering at the University of Genova.

We used R^2_{cv} for validation.

All the data were normalized between 0 and 1 before processing in the ANN. The ANN modules were implemented using MBP code. [7]

All the ANN simulations were done using SUN SPARCs from the Laboratory of Artificial Intelligence and Robotics (AIR-LAB) of the Department of Electronics and Information (DEI) of Milan Polytechnic and the mainframe SUN SPARC-Center 2000 at the EDP center of the same university.

We first made experiments using all the available descriptors and a fixed topology, increasing the number of neurons in the hidden layer. For each topology the algorithm started from 100 different points of the weight space to achieve the best performance avoiding local minima.

[6] The software is available through anonymous ftp on: risc6000.dibe.unige.it.

[7] MBP available with anonymous ftp at: risc6000.dibe.unige.it.

5.4. Rule extraction and qualitative prediction module

In a classical rule-based ES, knowledge is represented by production rules like:if < condition > then < action > .

Usually these rules are provided by human experts. Since we decided to use the new WHIM descriptors, there was no-one available to provide expertise through this representation of molecules. We had no other choice than to automatically extract rules.

We set a threshold for each data-set. All the compounds with a toxicity value below that threshold were considered non-active and the ones with higher values were considered active.

In order to extract rules, we used a well known machine learning program (the C4.5 by Quinlan [28]) and a statistical based, binary tree classifier (CART: classification and regression tree [29,30]).

For the qualitative prediction module, we needed an ES shell to use the automatically extracted KB. CLIPS [31,32] was chosen for this, because of its features of knowledge representation, portability, integration, interactive development, verify/validation, wide documentation and freely available software.

6. Results and discussion

6.1. Results for classification

The results on validation always indicated 100% correct classification. Since these results were ob-

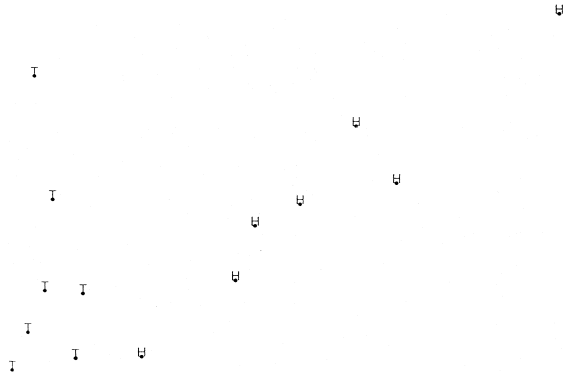

Fig. 2. Sammon's mapping of triazines and haloarenes.

Table 1
Results of experiments with 34 and 7 selected descriptors on triazines and haloarenes: modelling of phytotoxicity, *p*RB and *p*AHH

Number of neurons	Triazines		Haloarenes			
	Phytotoxicity		*p*RB		*p*AHH	
	34 Descriptors	7 Descriptors	34 Descriptors	7 Descriptors	34 Descriptors	7 Descriptors
	R^2_{cv}	R^2_{cv}	R^2_{cv}	R^2_{cv}	R^2_{cv}	R^2_{cv}
1	56.4	62.7	83.7	76.8	70.0	58.3
2	77.1	63.8	84.4	80.7	71.5	62.6
3	76.4	69.5	84.1	81.0	72.3	61.8
4	78.0	73.3	84.4	80.9	73.5	*62.7*
5	77.8	71.5	84.7	81.8	74.1	58.0
6	80.5	86.6	84.9	80.8	74.0	57.7
7	80.6	*87.7*	84.5	*82.8*	75.0	53.5
8	80.2	76.7	84.9	81.4	74.3	53.9
9	80.7	83.6	84.7	80.7	74.7	55.1
10	81.1	83.9	85.0	81.3	73.6	56.0
11	81.6	82.7	84.8	82.1	74.0	54.3
12	81.0	81.4	*85.4*	81.1	74.6	53.3
13	79.9	81.0	85.0	81.6	75.3	59.3
14	80.4	84.3	84.3	82.3	73.6	54.9
15	80.9		84.6		74.8	
16	80.2		84.4		75.7	
17	81.1		84.2		73.8	
18	80.2		84.1		74.6	
19	80.4		84.0		*76.8*	
20	81.1		84.3		75.0	
21	80.7		83.9		72.8	
22	*82.0*		83.3		75.2	
23	80.5		83.4		74.3	
24	80.5		83.6		74.2	

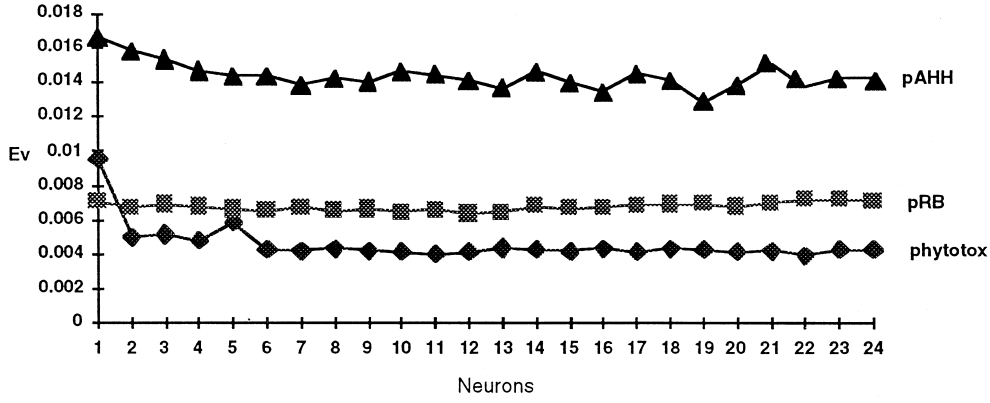

Fig. 3. E_v vs. number of neurons with 34 descriptors.

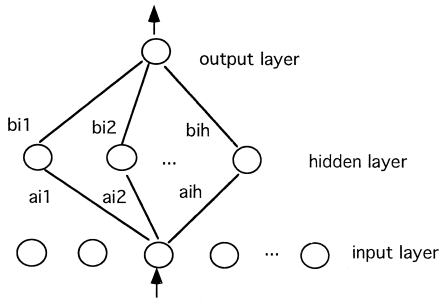

Fig. 4. Feed-forward NN for weight inspection.

tained with all possible combinations of the options, the robustness of the algorithm has been proved for this classification problem.

To show how the classes were distributed, we made a 2-D mapping of the compounds using Sammon's algorithm [24]. In Fig. 2, we can see that the two classes are very well distinguishable, and never overlap. The shape of the two clusters also shows that the class of triazines is much more homogeneous than the haloarenes.

6.2. Results with the ANN

Table 1 sets out the results of experiments with the ANN. Results with 34 descriptors were good. Increasing the number of neurons hardly improves the performance of the network (Fig. 3).

We thought we might possibly improve performance by reducing the number of descriptors so as to increase the ratio between the number of examples and number of features per example. To do this, we needed a method for inspecting the weights of the net and recognizing which of the inputs the algorithm really used. We virtually tried to open the 'black box' of the ANN to see what was happening inside.

Let us look at Fig. 4. For each input i to the net we calculated the function GMW_i (geometric mean for W_i):

$$GMW_i = \frac{\sum_{j=1}^{h} \sqrt{|a_{ij} \cdot b_{ij}|}}{h}.$$

If a_{ij} is high but the corresponding b_{ij} is small (or vice versa), the input i cannot propagate through the jth route to reach the final neuron. With GMW_i, we can see which signal i is really important in the net. What we did was to save the weights of the net when results were best and then calculate the geometric mean on these 'optimal' weights.

	T (Dim.)						A (Dim.)						G(Si)			K(Shape)						D(Dens.)						V (Vol.)						W	Ev	R2cv
Neu	u	m	v	e	p	s	u	m	v	e	p	s	u	m	s	u	m	v	e	p	s	u	m	v	e	p	s	u	m	v	e	p	s			
1	x		x					o						o	x		x				x	o	x		o	o			o	o				x	0.0096	56.4
2			x			o			o					o	x					x	x		x			x		o	o	o		o		x	0.0050	77.1
3								o	o	o	o	o							x	x	x		x	x		x		o			o			x	0.0052	76.4
4					x	o			o		o	o						o		x	x		x	x		x		o			o			x	0.0048	78.0
5	o						o		o	o		o							x	x	x		x			x		o	x		o		x	x	0.0049	77.8
6							o	x	o	o	o		o	o	o									x		x	x		x				x	x	0.0043	80.5
7								o	o	o	o			o						x	x		x	x		x				o	o		x	x	0.0042	80.6
8						o	o		o		o	o							x	x	x		x	x		x				o	o			x	0.0043	80.2
9	x				x			o		o	o		o								x		x	x		x		o		o	o			x	0.0042	80.7
10					x		o			o	o	o		o				o		x	x		x			x	x			o				x	0.0041	81.1
11						o			o	o			o	o					x		x		x	x		x	x	o				o		x	0.0040	81.6
12		o			x			o	o							o		o	x	x	x		x			x					o			x	0.0042	81.0
13				o			o		o	o	o						x	o		x	x		x			x					o		x	x	0.0044	79.9
14		o				o	o		o	o				o			x			x	x	x	x			x					o			x	0.0043	80.4
15					x		o		o	o	o					o	x			x	x		x			x					o	o		x	0.0042	80.9
16			x						o	o				o				o		x	x		x	x	o		o	o						x	0.0043	80.2
17				o					o	o	o		o	o			x				x		x	x	x	x				o				x	0.0041	81.1
18	o					o			o	o					x				x	x	x		x	x				o			o	o		x	0.0043	80.2
19					x				o		o	o		o					o	x	x		x			x					o		o	x	0.0043	80.4
20						o	o		o	o		o							o	x	x		x	x		x				o				x	0.0041	81.1
21			x		x		o		o	o	o						x	o		x	x		x					o				o		x	0.0042	80.7
22					x		o				o		o	o		o	x	o		x	x		x			x					o			x	**0.0039**	82.0
23			x			o	o		o				o						x	x	x	o	x			x				o		o		x	0.0043	80.5
24			x		x	o	o		o	o	o	o								x	x		x				x			o				x	0.0043	80.5

Fig. 5. Experiments for variables selection.

Table 2
Triazines predicted phytotoxicity on validation set using NN

#	Name	Predicted	Expected	|Err|
1	Mol 61	6.16	6.34	0.18
2	Mol 62	5.44	5.45	0.01
3	Mol 63	5.42	5.53	0.11
4	Mol 64	6.73	6.52	0.21
5	Mol 65	5.97	5.53	0.44
6	Mol 66	5.82	5.53	0.29
7	Mol 67	6.52	6.65	0.13
8	Mol 68	6.98	6.85	0.13
9	Mol 69	5.69	5.42	0.27
10	Mol 70	4.72	4.57	0.15
11	Mol 71	6.17	5.92	0.25
12	Mol 72	4.96	4.55	0.41
13	Mol 73	4.95	5.18	0.23
14	Mol 74	5.88	5.76	0.12

Our experiments are shown in Fig. 5. We made a list of all these means for all the inputs i and noted which of them was highest in all the simulations. We marked the seven strongest descriptors. Selecting the best descriptors from the tables as shown, we made new experiments with only seven descriptors, sometimes improving the performances of the models.

Table 3
Haloarenes predicted *p*RB on validation set using NN

#	Name	Predicted	Expected	|Err|
1	Mol 2	6.04	5.49	0.55
2	Mol 5	7.07	7.15	0.08
3	Mol 8	6.81	6.10	0.71
4	Mol 12	6.36	5.96	0.40
5	Mol 17	8.12	7.81	0.31
6	Mol 20	8.90	8.82	0.08
7	Mol 22	8.44	8.18	0.26
8	Mol 25	8.49	8.83	0.34
9	Mol 29	3.80	4.38	0.58
10	Mol 32	4.68	3.61	1.07
11	Mol 35	5.19	4.07	1.12
12	Mol 39	5.94	6.46	0.52
13	Mol 41	6.39	6.66	0.27
14	Mol 44	5.76	6.40	0.64
15	Mol 47	6.36	6.70	0.34
16	Mol 50	4.64	4.70	0.06
17	Mol 53	6.13	5.89	0.24
18	Mol 57	5.94	5.08	0.86
19	Mol 61	3.62	3.85	0.23
20	Mol 64	4.66	5.39	0.73
21	Mol 67	6.14	6.92	0.78
22	Mol 69	4.53	5.15	0.62
23	Mol 71	4.93	4.80	0.13

Table 4
Haloarenes predicted *p*AHH on validation set using NN

#	Name	Predicted	Expected	|Err|
1	Mol 2	5.89	4.00	1.89
2	Mol 5	6.65	6.44	0.21
3	Mol 8	7.06	6.23	0.83
4	Mol 12	7.37	7.68	0.31
5	Mol 17	6.60	7.38	0.78
6	Mol 20	10.23	9.26	0.97
7	Mol 22	9.64	9.19	0.45
8	Mol 25	9.03	10.38	1.35
9	Mol 32	4.36	4.21	0.15
10	Mol 35	5.30	4.71	0.59
11	Mol 39	6.67	5.88	0.79
12	Mol 41	7.18	5.98	1.20
13	Mol 44	5.76	7.07	1.31
14	Mol 47	7.48	8.10	0.62
15	Mol 50	6.12	7.42	1.30
16	Mol 53	7.21	6.97	0.24
17	Mol 57	6.94	7.37	0.43
18	Mol 61	3.15	3.00	0.15
19	Mol 64	5.46	6.01	0.55
20	Mol 67	8.25	9.62	1.37
21	Mol 69	5.57	5.68	0.11
22	Mol 71	5.99	4.89	1.10

Comparing the results with 34 and 7 descriptors (Table 1), we see that the reduction in the number of descriptors increased the performance of the models for triazines (R^2_{cv} goes from 82.0% to 87.7%), slightly decreased the *p*RB models (R^2_{cv} goes from 85.4% to 82.8%) and strongly decreased the *p*AHH ones (R^2_{cv} goes from 76.8% to 62.7%).

To test the reliability of the models, Tables 2–4 show the net's answers when prompted with the validation set of compounds, and compare the results with the expected ones; for triazines the mean error is 0.21, for haloarenes (*p*RB) 0.47 and for *p*AHH 0.76.

6.3. Comparisons of the obtained models

It is difficult to make a useful comparison with the results found in the literature, mainly because of different validation methods. Nevertheless, to give an idea of the difficulty of modelling toxicity, we report the results found on papers dealing with the same sets of compounds.

A triazines data-set was used in Ref. [33]. At the beginning, the set comprised 78 compounds, but was

Table 5
Percentage of *x*-validated error on rule extraction with CART and C4.5 obtained with leave-one-out method

	Triazines		Haloarenes			
	Phytotoxicity		*p*RB		*p*AHH	
	34 Descriptors	7 Descriptors	34 Descriptors	7 Descriptors	34 Descriptors	7 Descriptors
C4.5	*24.30*	31.10	28.20	32.40	30.40	26.10
CART	27.02	32.43	*21.13*	28.17	*21.74*	26.09

then reduced to 71 because of seven outliers. The authors gave up the model at that stage.

In Ref. [34] we find the results for haloarenes. Sulea et al. [34] used topological descriptors and various combinations of subsets of the original data-set; they obtained these results, with the leave-one-out method for validation:

*p*RB: $R^2_{cv} = 59\%$

*p*AHH: $R^2_{cv} = 55\%$.

6.4. The qualitative prediction module

As we have already explained, this module has to assign an active/non-active class to the compound and explain its conclusion. The information regarding toxicology is low, but the real information here is *why* a result was obtained.

In order to extract rules, we used C4.5 and CART. For both algorithms we first tried 34 descriptors and then with the seven descriptors selected with the ANN approach. The leave-one-out method was used for validation. We could thus compare these two algorithms properly. In Table 5 we show the best results for the rules extracted by both. As we can see, C4.5 was better than CART for phytotoxicity classification, while CART performed well for haloarenes. The rules extracted with just seven descriptors did not perform badly, but could not outperform the ones extracted with all the available descriptors.

7. Conclusions

The prototype [8] was implemented following three principles: modularity, portability and transparency. To achieve these we propose a modular, easily expandable architecture. We used a strongly standardized language like ANSI-C, and whenever possible public domain software.

We used the LVQ NN for classification, obtaining a 100% correct assignment. The quantitative prediction was obtained with trained ANN. For triazines the net dimensions were $7 \times 7 \times 1$ and the predictive power 87.7%; for haloarenes, referring to the prediction of *p*RB the net dimensions were $34 \times 12 \times 1$ and the predictive power 85.4%; for the *p*AHH activity, the net dimensions were $34 \times 19 \times 1$ and the predictive power 76.8%.

The qualitative module was implemented with CLIPS as inference engine; the rules extracted with CART are four for triazines and gave a predictive power of 76%; for *p*RB there were four rules and the predictive power was 79%; for *p*AHH there were five rules and the predictive power was 78%.

Here CLIPS was definitely under-used, since KB were usually made of just four or five rules and we

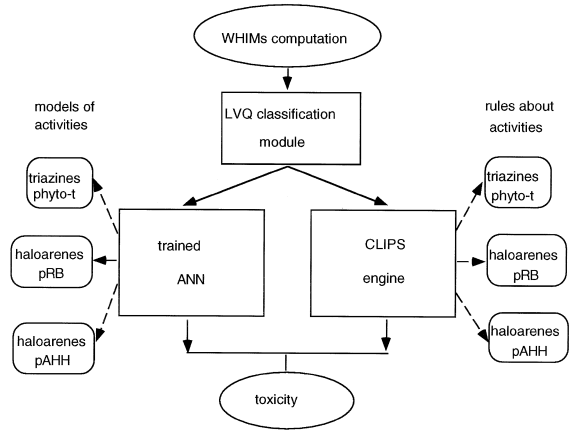

Fig. 6. Final architecture of hybrid toxicology expert (HyTEx) prototype.

[8] The prototype has been implemented on a 486 DX-4 100 MHz with 16 M of RAM memory and running Linux.

never had conflict of resolution problems. CLIPS was anyway used to integrate more and much wider KB in the future. A complete architecture of the prototype is shown in Fig. 6.

This new architecture has not solved all the problems related to ES in toxicology. We believe that the ultimate solution would be to create a hybrid system approaching the problem from all possible angles: considering the metabolic decay, the structural alerts in residues, the molecule as a whole and as a set of different substructures. Biochemical mechanisms must not be neglected either. We are still far from this target, mainly because of the lack of databases with enough coherent information. This work is just one of the several steps needed to achieve these goals.

At this stage, only two classes of chemicals can be recognized by this prototype. Because of HyTEx's modular structure, we can easily add new classes of molecules. A crucial point is always the classification module. If needed, this can be improved with a multi-level classifier.

The HyTEx prototype can also be used as part of a more complex system considering metabolic decay, if required for every metabolite. When prompted with a compound whose class is recognizable by the system, HyTEx gives good results; similarly, the quantitative and qualitative modules were successful.

Acknowledgements

G.G. and E.B. gratefully acknowledge partial financial support from the EU Copernicus EST project, under contract CP94 1029.

References

[1] J. Doull, C.D. Klaassen, M.O. Amdur (Eds.), Casarett and Doull's Toxicology: The Basic Science of Poisons, 2nd edn., Macmillan, New York, NY, 1980, p. 778.
[2] G.S. Omenn, Toxicology 102 (1995) 23–28.
[3] M. Nendza, J. Volmer, W. Klein, in: W. Kalcher, J. Devillers (Eds.), Practical Applications of Quantitative Structure–Activity Relationships (QSAR) in Environmental Chemistry and Toxicology, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1990, 213–240.
[4] J. Ashby, D. Paton, Mutation Research 286 (1993) 3–74.
[5] HazardExpert, from CompuDrug Chemistry, H-1395 Budapest 62, P.O.B. 405.
[6] M. Nakadate, M. Hayashi, T. Sofuni, E. Kamata, Y. Aida, T. Osada, T. Ishibe, Y. Sakamura, M. Ishidate Jr., Environmental Health Perspectives 96 (1991) 77–89.
[7] K. Enslein, V.K. Gombar, B.W. Blake, Mutation Research 305 (1993) 47–61.
[8] V.K. Gombar, K. Enslein, B.W. Blake, Mutation Research 302 (1993) 7–12.
[9] G. Klopman, Journal of the American Chemical Society 24 (1984) 7315–7321.
[10] H.S. Rosenkranz, G. Klopman, Teratogenesis, Carcinogenesis and Mutagenesis 10 (1990) 73–88.
[11] H.S. Rosenkranz, G. Klopman, Mutagenesis 4 (1990) 333–361.
[12] H.S. Rosenkranz, G. Klopman, Mutation Research 232 (1990) 249–260.
[13] G. Klopman, H.S. Rosenkranz, Environmental Health Perspectives 96 (1991) 67–75.
[14] G. Klopman, H.S. Rosenkranz, Mutation Research 305 (1993) 33–46.
[15] D.F.V. Lewis, C. Ioannides, D.V. Parke, Mutation Research 291 (1993) 61–77.
[16] E. Benfenati, G. Gini, Toxicology 119 (1997) 213–225.
[17] J.C. Dearden, Physico-chemical descriptors, in: Environmental Chemistry and Toxicology, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1990, 25–59.
[18] A. Sabljic, in: W. Kalcher, J. Devillers (Eds.), Practical Applications of Quantitative Structure–Activity Relationships (QSAR), in Environmental Chemistry and Toxicology, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1990, 61–82.
[19] R. Todeschini, P. Gramatica, R. Provenzani, E. Marengo, Chemometrics and Intelligent Laboratory Systems 27 (1995) 221–229.
[20] R. Todeschini, P. Gramatica, Quant. Struct.–Act. Relatsh. 16 (1997) 113–119.
[21] R. Todeschini, P. Gramatica, Quant. Struct.–Act. Relatsh. 16 (1997) 120–125.
[22] R. Todeschini, M. Vighi, R. Provenzani, A. Finizio, P. Gramatica, Chemosphere 8 (1995) 1527–1545.
[23] R. Todeschini, WHIM-3D/QSAR—Software for the calculation of the WHIM descriptors, Release 2.1 for Windows, Talete, Milan, Italy, 1996.
[24] T. Kohonen, Self-organizing maps, Springer Series in Information Sciences, Springer-Verlag, Berlin, 1995.
[25] T. Kohonen, J. Hynninen, J. Kangas, J. Laaksonen, K. Torkkola, LVQ_PAK: The Learning Vector Quantization Program Package, prepared by the LVQ Programming Team of the Helsinky University of Technology, Laboratory of Computer and Information Science, Finland, 1995.
[26] D. Anguita, Matrix Back Propagation v1.1: User's Manual, 1993, downloadable from: risc6000.dibe.unige.it.
[27] D. Anguita, M. Pampolini, G. Parodi, R. Zunino, Proceedings of the ICANN 93, September 13–16, 1993, Amsterdam, p. 500.
[28] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[29] L. Breiman, J.H. Friedman, R.A. Olshen, C.J. Stone, Classification and Regression Trees, Wadsworth, Belmont, CA, 1984.
[30] SCAN: Software for Chemometric Analysis: Reference Manual, Release 1 for Windows, Minitab, 1995, information available in: http://jsc.nasa.gov.
[31] CLIPS Reference Manual, Vol. I: The Basic Programming Guide; Vol. II: The Advanced Programming Guide; Vol. III: The Interfaces Guide, 1993. NASA Software Technology Branch, Lyndon B. Johnson Space Center, information available in: http://jsc.nasa.gov.
[32] J.C. Giarratano, CLIPS v6.0: User's Guide, NASA Software Technology Branch, Lyndon B. Johnson Space Center, 1993.
[33] M.L. Tosato, S. Marchini, L. Passerini, D. Cesareo, D. Bonelli, G. Cruciani, S. Clementi, Chimica Oggi 1988 (1988) 55–59.
[34] T. Sulea, L. Kurunczi, Z. Simon, SAR and QSAR in Environmental Research 3 (1995) 37–61.