Predicting Physical−Chemical Properties of Compounds from Molecular Structures by Recursive Neural Networks

Luca Bernazzani,[†] Celia Duce,[†] Alessio Micheli,[‡] Vincenzo Mollica,[†] Alessandro Sperduti,[§]
Antonina Starita,[‡] and Maria Rosaria Tiné*,[†]

Dipartimento di Chimica e Chimica Industriale, Università di Pisa, Via Risorgimento 35, I-56126 Pisa, Italy, Dipartimento di Informatica, Università di Pisa, Largo B. Pontecorvo 3, I-56127 Pisa, Italy, and Dipartimento di Matematica Pura ed Applicata, Università di Padova, Via Belzoni 7, I-35131 Padova, Italy

Received March 22, 2006

In this paper, we report on the potential of a recently developed neural network for structures applied to the prediction of physical chemical properties of compounds. The proposed recursive neural network (RecNN) model is able to directly take as input a structured representation of the molecule and to model a direct and adaptive relationship between the molecular structure and target property. Therefore, it combines in a learning system the flexibility and general advantages of a neural network model with the representational power of a structured domain. As a result, a completely new approach to quantitative structure−activity relationship/quantitative structure−property relationship (QSPR/QSAR) analysis is obtained. An original representation of the molecular structures has been developed accounting for both the occurrence of specific atoms/groups and the topological relationships among them. Gibbs free energy of solvation in water, $\Delta_{\mathrm{solv}}G°$, has been chosen as a benchmark for the model. The different approaches proposed in the literature for the prediction of this property have been reconsidered from a general perspective. The advantages of RecNN as a suitable tool for the automatization of fundamental parts of the QSPR/QSAR analysis have been highlighted. The RecNN model has been applied to the analysis of the $\Delta_{\mathrm{solv}}G°$ in water of 138 monofunctional acyclic organic compounds and tested on an external data set of 33 compounds. As a result of the statistical analysis, we obtained, for the predictive accuracy estimated on the test set, correlation coefficient $R = 0.9985$, standard deviation $S = 0.68$ kJ mol^{-1}, and mean absolute error MAE $= 0.46$ kJ mol^{-1}. The inherent ability of RecNN to abstract chemical knowledge through the adaptive learning process has been investigated by principal components analysis of the internal representations computed by the network. It has been found that the model recognizes the chemical compounds on the basis of a nontrivial combination of their chemical structure and target property.

INTRODUCTION

To predict the physical−chemical properties of compounds, starting from the molecular structure, is a challenging research objective, and many efforts have been spent over time in the development of predictive methods. In recent years, various machine-learning techniques, such as artificial neural networks (NN) and genetic algorithms, have been applied to the formulation of quantitative structure−activity or quantitative structure−property relationships (QSAR/QSPR). Neural networks are universal approximators able to learn, from a set of examples, nonlinear relationships between a proper representation of a chemical structure and a given target property. In standard NN approaches, the structure of a molecule is described by a set of structural or chemical parameters (molecular descriptors). More recently, a model based on recursive neural networks (RecNN) has been proposed for QSPR/QSAR[1−4] which is able to deal directly with structured domains. The possibility of processing structured information using neural networks is appealing in the context of prediction tasks in chemistry, where, on one hand, the compounds can naturally be represented as labeled graphs and, on the other, the choice of suitable molecular descriptors is a difficult and time-consuming charge. The RecNN model was successfully applied to the prediction of the boiling points of linear and branched alkanes[1,2] and of the pharmacological activity of a series of substituted benzodiazepines.[1,3,4] In both cases, the molecules were represented as labeled chemical graphs, but the representation rules of their structures were specifically defined for the single class of compounds taken into consideration.

In the present report, we intend to address more general chemical tasks dealing with a wider set of compounds. Accordingly, we propose a rational approach to the representation of chemical structures by using a limited number of fundamental atomic groups ordered as the corresponding two-dimensional molecular graph. As a first application, we employed this approach for the prediction of the standard Gibbs free energy of solvation in water, $\Delta_{\mathrm{solv}}G°$, of a set of monofunctional compounds. Solvation free energies were selected as the target property because of the availability of a large and reliable data set. Indeed, a homogeneous and critically reviewed database is needed in order to assess

* Corresponding author tel.: +39-050-2219311; fax: +39-050-2219260; e-mail: mrt@dcci.unipi.it.

[†] Dipartimento di Chimica e Chimica Industriale, Università di Pisa.

[‡] Dipartimento di Informatica, Università di Pisa.

[§] Università di Padova.

10.1021/ci060104e CCC: $33.50
Published on Web 09/09/2006 PAGE EST: 12.1

which performances may arise from the application of the proposed model to a given problem. Moreover, in the literature, many different approaches, exploiting both the computational chemistry and QSAR/QSPR fields, deal with the problem of predicting the $\Delta_{\text{solv}}G°$ values or other related quantities. A straightforward comparison between our results and those obtained with different methods is then possible, allowing for an authoritative validation of the proposed model on a significant benchmark. Finally, because the structure features which determine the free energy of solvation of a molecule have been thoroughly analyzed, the choice of $\Delta_{\text{solv}}G°$ as the target property turns out to be the most proper one in order to extract the chemical knowledge learned by the RecNN model during the training process.

PREDICTION OF THE SOLVATION FREE ENERGY

The standard free energy of solvation in water represents the Gibbs energy change in the isothermal transfer of a solute molecule from the ideal gas phase to an aqueous solution in the chosen standard states, the most convenient being 1 mol dm^{-3} concentration in both phases.[5] Different approaches have been proposed in the literature for predicting $\Delta_{\text{solv}}G°$. They can be classified as theory-based computational chemistry approaches and QSAR/QSPR approaches. Usually, the theory-based models directly compute $\Delta_{\text{solv}}G°$, whereas in the empirical field of QSAR/QSPR, other related quantities such as the logarithm of the Ostwald solubility coefficient, log L_W, or of the Henry constant, log H, are usually the target properties because of a more intuitive link to the solubility or partitioning property. Below, a comparative analysis of the distinctive features of each approach will be performed. An overview of the main methods proposed in the literature is summarized in Table 1.

Theory-Based Computational Chemistry Approaches. The principal computational chemistry approaches for computing solvation free energies account for the solvent as either a continuum medium or a large number of discrete molecules.[6−21]

The continuum-model-based approach[7−10] describes the solute at the quantum mechanical level in the reaction field of the solvent considered as a continuum dielectric. It provides an evaluation of the free energies of solvation by adding, to the electrostatic term, contributions accounting for the formation in the solvent of a cavity able to lodge the solute molecule and for solute−solvent repulsive and dispersive interactions. The procedure requires the assessment of the charge distribution for the solute and of different parameters for describing the solute's cavity, the nonelectrostatic terms, and the permittivity inside and outside the cavity. In this framework, different models have been applied to the calculation of $\Delta_{\text{solv}}G°$ by Barone et al.,[12] Klamt et al.,[13] and the group of Cramer and Truhlar.[14]

The discrete models use Monte Carlo (MC) statistical mechanics[17] or molecular dynamics simulations[18,19] to model the solute and solvent. In both cases, the condensed system is represented by an assembly of interacting particles: the statistical distribution of any property, or its evolution in time, is obtained as a sum over all particles with appropriate rules. These techniques have been used by Duffy and Jorgensen[20] and by Murray and co-workers[21] to correlate the $\Delta_{\text{solv}}G°$ of different solutes in water to MC simulation-derived and molecular electrostatic potential (MEP)-derived descriptors.

The above-mentioned theoretical methods allowed the derivation of rather accurate values of solvation free energies. Their most appealing advantage with respect to the QSAR/QSPR approaches is that they are able to provide a better understanding of the physical meaning of all the factors contributing to the solvation process. Unfortunately, these approaches are time-consuming and are mainly applied to monofunctional molecules of small dimensions.

Standard QSAR/QSPR Approaches. A quite complete review of the standard QSAR/QSPR methods is already reported in the literature.[22−24] Below, we reconsider the different QSPR/QSAR approaches in a general perspective and according to their decomposition in subtasks. This should help us to analyze the advantages of our approach, as we can prove that the model we propose can be a suitable tool for the automatization of fundamental parts of the QSPR/QSAR analysis.

The basic idea of a QSPR/QSAR study is to find an appropriate function F that predicts any molecular property (QSPR), or the biological activity (QSAR), using information related to only the molecular structure:

$$\text{Property} = F(\text{Structure}) \tag{1}$$

The input domain of F is a set of molecular structures, where the term "structure" refers to global information characterizing the molecule (molecular shape, chemical functionalities, etc.), and the output domain is typically a set of real numbers, which are used to quantify the property of interest. Hence, the function F can be seen as a functional transduction defined on a structured domain.

For the sake of a detailed and uniform view of the different aspects of the various approaches, the function F can be decomposed into functions that are more specific. This corresponds to the definition of a *feature representation* function f and of a *mapping* function g. As already outlined,[3,4] the function f, in turn, entails the representation of the molecular structure (through the function f_R) and the subsequent *encoding* of the structure into a set of numerical descriptors (through the function f_E).[4]

An either linear or nonlinear regression model can be used to compute the output value, realizing a mapping function g from the descriptor space to the physical chemical property.

According to this view, F can be decomposed as follows:

$$F(\cdot) = g[f(\cdot)] \tag{2}$$

and further

$$f(\cdot) = f_E[f_R(\cdot)] \tag{3}$$

Different approaches have been used to realize the f and g functions. In this view, the choice of the functions f and g is the discriminant aspect among the different approaches, with a major role of the issues related to the f function. In fact, three families of methods can be identified on the basis of the choice of the function f: methods based on molecular properties using experimental quantities as descriptors, methods of group contribution, and methods employing structural molecular descriptors.

To the *molecular-property-based methods* belong the general linear solvation energy relationships (LSER) originally proposed by the group of Kamlet and Taft[25,26] and

Table 1. Overview of Main Methods Employed for the Prediction of Free Energies of Solvation and Related Quantities

reference	model	data set description[a]	statistics[b]
Theory-Based Computational Chemistry Approaches			
Barone et al.[12]	PCM	43 neutral solutes and 27 ions; most important functional groups in monofunctional open-chain molecules + pyridine, pyrrole, and differently substituted benzene rings	$S = 0.88$; MAE = 0.67 or $S = 2.30$; MAE = 1.80 (depending on the normalization procedure)
Klamt et al.[13]	COSMO−RS	217 molecules and six properties, including $\Delta_{\mathrm{solv}}G°$ in water; altogether, about 642 data points (163 $\Delta_{\mathrm{solv}}G°$ data)	$S = 1.55$ (on $\Delta_{\mathrm{solv}}G°$ data)
Cramer and Truhlar[14]	SM6	273 organic compounds, 112 ions, and 31 ion−water clusters	MAE = 2.6 (on 273 organic compounds)
Duffy and Jorgensen[20]	discrete model based on MC-derived descriptors	85 monofunctional compounds in water ($C \le 6$) and some aromatic and cyclic compounds	$R^2 = 0.89$; $S = 2.80$; MAE = 2.25
Murray et al.[21]	MEP-derived descriptors	47 monofunctional compounds including some aromatic, cyclic, and etheroaromatic compounds, and three bifunctional compounds	$R^2 = 0.988$; $S = 1.57$; MAE = 1.14
Molecular-Property-Based Methods			
Abraham et al.[27]	LSER	training set of 408 chemicals (log L_W)	$R = 0.9976$; $S = 0.86$
Group Contributions Methods			
Hine and Mookerjee[30]	group contributions; bond contributions	(a) 212 log L_W data; (b) 263 log L_W data	(a) $S = 0.69$; (b) $S = 2.40$
Meylan and Howard[31]	bond contributions	training set of 345 log L_W of organic compounds; test set of 74 log L_W of organic compounds	training: $S = 1.94$; MAE = 1.20 test: $S = 2.63$; MAE = 1.77
Cabani et al.[32]	group contributions	350 noncharged organic compounds[c]	$S = 0.51$ (on a subset of 209 monofunctional compounds)
Wendoloski et al.[33]	HLOGS/ALOGS	training set of 265 organic molecules; test set of 27 organic molecules	training: $R^2 = 0.941$, $S = 2.43$ (HLOGS); $R^2 = 0.960$, $S = 1.59$ (ALOGS) test: $R = 0.96$; $S = 3.60$ (ALOGS)
Hou et al.[35]	group contributions based on SASA model	377 neutral molecules	$S = 1.92$; MAE = 2.13
Molecular-Structure-Based Methods			
Nirmalakhandan and Speece[38]	three structure-based molecular descriptors	training set of 267 organic molecules (log H); test set of 175 organic molecules (log H)	training: $R^2 = 0.98$; $S = 2.05$ test: $R^2 = 0.95$
Russel et al.[39]	five structure-based molecular descriptors	training set of 63 organic molecules (log H); test set of nine organic molecules (log H)	training: MAE = 2.11 test: MAE = 1.94
Katritzky et al.[44]	CODESSA	training set of 408 chemicals (log L_W)	$R^2 = 0.942$; $S = 2.97$; MAE = 2.40
English and Carroll[47]	two feed-forward neural network architectures (a, b)	training set of 303 organic molecules (log H); test set of 54 organic molecules (log H)	training: (a) $R^2 = 0.987$, $S = 1.28$; (b) $R^2 = 0.99$, $S = 1.15$ test: (a) $R^2 = 0.979$, $S = 1.60$; (b) $R^2 = 0.985$, $S = 1.35$
Yaffe et al.[24]	(a) cognitive classifier Fuzzy ARTMAP b) back-propagation for neural networks	training set of 421 organic molecules (log H); test set of 74 organic molecules (log H)	training: (a) $S = 0.06$, MAE =0.06; (b) $S = 1.54$, MAE =1.65 test: (a) $S = 0.68$, MAE =0.74 (b) $S = 1.34$, MAE =1.37

[a] $\Delta_{\mathrm{solv}}G°$ in water if not otherwise specified. [b] Standard deviation, S, and mean absolute error, MAE, in kJ mol^{-1}; R, linear correlation coefficient. [c] The same approach is also applied to 197 values of $\Delta_{\mathrm{solv}}H°$, 272 values of $\bar{C}^{o}_{p,2}$, and 425 values of $\bar{V}^{o}_{2}$ in water.

improved by Abraham and co-workers.[27−29] In this approach, the feature representation of the molecule (function *f*) is realized through several characteristic experimental properties (solvatochromic parameters), while a multilinear regression analysis (MLR) is employed as the mapping function *g*.[27] The ability of LSER descriptors to make a priori predictions is limited because a fixed set of experimentally determined values is required for each compound.

The *group contribution* (GC) *methods* rely on the basic idea that a solute molecule acts as a number of fragments (atoms, bonds, chemically significant groups, and larger molecular fragments) independently contributing to the investigated property. The general equation commonly employed in this additivity scheme is

$$Y = \sum_j n_j B_j \tag{4}$$

where Y is the thermodynamic function of interest and B_j is the contribution to the property by the jth group present n_j times in the solute structure. The values of the group contributions are usually determined by MLR analysis through eq 4. This relationship can be written also as

$$Y = \mathbf{N}\cdot\mathbf{B} \tag{5}$$

where **N** is the row vector of the group frequencies and **B** the column vector of the group contributions. As it can be easily recognized, in the group contribution approach, the function f_R consists of extracting from the molecular formula the fragments the molecule should be divided into and the frequency of their occurrence. On the other hand, f_E is the construction of the row vector of the frequencies, by putting each frequency in the correct position univocally identifying each group. The mapping function g is simply the product of the row vector of the frequencies and the column vector of the group contributions. GC methods have been proposed by Hine and Mookerjee,[30] Meylan and Howard,[31] Cabani and co-workers,[32] and Wendoloski and co-workers.[33] A further class of GC methods, based on solvent accessible surface areas (SASAs), has been proposed by Eisenberg and Malachlan[34] and improved by Hou et al.[35]

In the *molecular-structure-based methods*, molecular descriptors such as topological indices, quantum-chemical descriptors, geometrical and electrostatic descriptors, and so forth are used to encode (function f) the molecules. These methods were applied by Nirmalakhandan and co-workers[36−38] and by Russel and co-workers.[39] In both cases, a linear regression analysis is used to realize the mapping function g.

The definition/selection of proper molecular descriptors is a difficult task, and furthermore, it is target-dependent. This limit can be partially overcome by methods based on automatic feature selection.[22−24,40−43] Starting from a very large set of theoretical descriptors, feature selection methods are aimed at automatically selecting the most suitable features for the prediction of a given property. The CODESSA program[42] developed by Katritzky calculates all of the most important known structure-based descriptors and, by using a heuristic procedure, also selects the fixed-size MLR model which provides the best statistical performance parameters. CODESSA PRO has been successfully applied to a large variety of problems.[22,23,44−46]

In recent years, various approaches have been taken into consideration to realize the function g by more complex machine-learning models such as the neural networks.[24,41,47] NNs, in fact, are powerful data modeling tools able to approximate nonlinear relationships among chemical structural parameters and physical−chemical properties. NNs/QSPR models for estimating the Henry constant in water were recently reported by English and Carroll.[47] More recently, QSPR methods have been proposed which couple a feature selection approach with a nonlinear mapping function. These methods have been applied to the prediction of the Henry's law constant and the solubility of organic compounds in water by Yaffe and co-workers[24] and by Mitchell and Jurs,[41] respectively.

It must be pointed out that the nonlinear variable selection, for nonlinear g models, depends on the chosen model and still constitutes an ongoing issue of research. Moreover, all of these approaches use fixed-size numerical vectors as input to the regression function g, and they are not meant for dealing with structured domains. In other words, as the previously described methods, they rely on a feature representation function f for the molecules returning predetermined numerical descriptors. Hence, besides the use of a powerful mapping function, these approaches do not introduce any methodological novelty in the handling of the molecular structure.

Nonstandard QSAR/QSPR Approaches. From the analysis of the previous approaches, it results that major benefits can be introduced in the QSAR/QSPR approaches by a simultaneous learning of the encoding function f and mapping function g. As discussed, a relevant direction to tackle this problem is partially obtained by methods based on feature selection using a measure of the global F function performance as the objective function for heuristic selection. However, by construction, selection methods are based on the occurrence of relevant features in the initial set of descriptors. In particular, such approaches cannot consider innovative structural features or descriptors not included in the initial set.

A more general and appealing approach can be to *generate* specific descriptors for the regression task to be solved. To this aim, we use RecNN methods, which belong to the area of machine-learning models developed to directly handle structured data. The main advantage of the RecNN approach stems from the use of the learning for the construction, or encoding, of specific descriptors. In particular, the encoding function of the molecular structures f_E is learned together with the regression function g. A second important point concerns the treatment of molecules as varying size structures. RecNN allows taking directly as input labeled structures of variable size, that is, a hierarchical set of labeled vertexes connected by edges belonging to subclasses of graphs, such as rooted trees. Labeled structures are high abstract and graphical tools that can represent a molecule at different levels of detail, such as atoms, bonds, or chemical groups. A natural representation of a molecule is made possible by reproducing its 2D structure in the input graph. To this aim, the function f_R is used as a tool to model molecules as structured data.

In this paper, we report the use of RecNNs to describe the standard free energy of solvation, $\Delta_{solv}G°$, in water of a set of 179 acyclic monofunctional organic compounds. The rules used to represent the molecules examined in this work in the form of labeled rooted ordered trees will be presented and discussed in the next section.

THE RECURSIVE NEURAL NETWORK MODEL

In this section, we present the approach based on recursive neural networks for the processing of structured domains.[1−43,48,49] First, we provide a proper instantiation of the input and output domains of the functions f_E and g implemented by the RecNN.

Let the structured input domain for f_E, denoted by G, be a set of labeled directed positional acyclic graphs (DPAGs). In a DPAG, for each vertex (or *node*), a total order on the edges leaving from it is defined and a position is assigned to each edge.

Moreover, let us assume that G has for each node a bounded out-degree and that each DPAG possesses a supersource, that is, a vertex s such that every vertex in the graph can be reached by a directed path starting from s. *Labels* are tuples of variables and are attached to vertexes. Let $\mathscr{R}^n$ denote the label space.

Here, we consider a subclass of DPAGs formed by prohibiting cycles in the undirected skeleton of the graph, the set of the k-ary trees. In the case of trees, the supersource is defined by its *root node*. *k-ary trees* (*trees* in the following)

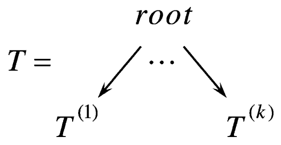


Figure 1. *k*-ary tree *T* rooted in the node *root*, with subtrees $T^{(1)}$,..., $T^{(k)}$.

are rooted positional trees with a finite out-degree *k*. This class of structures includes the class of rooted ordered trees and, clearly, sequential and vectorial data.

Given a node v in the tree $T \in G$, we give the following definitions: the children of v are the node successors of v, each with a position $j = 1, ..., k$; k is the maximum out-degree over G, that is, the maximum number of children for each node; $L(v)$ in $\mathcal{R}^n$ is the input label associated with v, and $L_i(v)$ is the ith element of the label; the *subtree* $T^{(j)}$ is a tree rooted at the jth children of v.

Vertexes with a zero out-degree are *leaves* of the tree. The set of external vertexes is the *frontier*. *Traversal* of a tree allows for systematically visiting (and processing) all of the nodes of the tree in some order: in particular, processing in a recursive manner all subtrees and finally the root, we define a *postorder* traversal. The scheme of a *k*-ary tree is reported in Figure 1.

The descriptor (or code) space is chosen as $\mathcal{R}^m$, while the output space, for our purpose, is defined as $\mathcal{R}$. Finally, the class of functions which can be realized by a RecNN can be characterized as the class of functional graph transductions described in the form $g[f_E(\cdot)]$, where $f_E(\cdot)$: $G \rightarrow \mathcal{R}^m$ is the encoding function and $g(\cdot)$: $\mathcal{R}^m \rightarrow \mathcal{R}$ the output function.

The functions and domains involved in the definition of the RecNN are shown in eq 6.

$$\text{Structure} \xrightarrow[f_R(\,)]{} G \xrightarrow[f_E(\,)]{} \mathcal{R}^m \xrightarrow[g(\,)]{} \mathcal{R} \qquad (6)$$

In our approach, we define a function f_E that allows the progressive encoding of an input structure, for example, a tree, using at each step a neural computational model τ_{NN}. The function τ_{NN} is used to process each node of a given structure. Given a node in a tree T, τ_{NN} uses the information available at the current node, (1) the numerical label attached to the node (in $\mathcal{R}^n$) and (2) the numerical code for each subgraph of the node (in $\mathcal{R}^m$), and produces a code in $\mathcal{R}^m$. As a result, if k is the maximum out-degree of T in G, τ_{NN} is defined as

$$\tau_{NN} : \mathcal{R}^n \times \underbrace{\mathcal{R}^m \times ... \times \mathcal{R}^m}_{k} \rightarrow \mathcal{R}^m \qquad (7)$$

The information coded in the k spaces $\mathcal{R}^m$ is the new part introduced in the RecNN model. We show in the following that these extra inputs to the model allow the memorization of structural information and make the model responsive to the topological structure of the input compound. First of all, let us consider, for example, the simplest nonlinear realization for τ_{NN} that uses a single recursive neural unit ($m = 1$). Given a node v of T, the output X in $\mathcal{R}$ of the recursive neuron (i.e., the code of v), is computed as follows:

$$X = \tau_{NN}[L(v), X^{(1)}, ..., X^{(k)}] = \phi\left(\sum_{i=1}^{n} w_i\, L_i(v) + \sum_{j=1}^{k} \hat{w}^j X^{(j)} + \theta\right) = \phi\left(\mathbf{W}L + \sum_{j=1}^{k} \hat{w}^j X^{(j)} + \theta\right) \qquad (8)$$

where ϕ is a nonlinear sigmoidal function, $L(\nu)$ in $\mathcal{R}^n$ is the label of v, θ is the bias term, $\mathbf{W}$ in $\mathcal{R}^n$ is the weight vector associated with the label space, $X^{(j)}$ in $\mathcal{R}$ is the code for the jth subgraph (subtree) of v, and $\hat{w}^j$ in $\mathcal{R}$ is the weight associated with the jth subgraph space. When $m > 1$, we obtain a neural network (with m units):

$$\mathbf{x} = \tau_{NN}[L(\nu), \mathbf{x}^{(1)}, ..., \mathbf{x}^{(k)}] = \Phi\left(\mathbf{W}L + \sum_{j=1}^{k} \hat{\mathbf{W}}^j \mathbf{x}^{(j)} + \theta\right) \qquad (9)$$

where $\mathbf{x}$ is a vector in $\mathcal{R}^m$, Φ is a set of m sigmoidal functions, θ in $\mathcal{R}^m$ is the bias vector, $\mathbf{W}$ in $\mathcal{R}^{m\times n}$ is the weight matrix associated with the label space, $\mathbf{x}^{(j)}$ in $\mathcal{R}^m$ are the vectorial codes for each subgraph (subtree) of the current node, and $\bar{\mathbf{W}}^j$ in $\mathcal{R}^{m\times n}$ is the weight matrix associated with the jth subgraph space.

The composition of τ_{NN} used to encode a structured set of nodes, for example, a tree T as shown in Figure 1, is defined by the following *recursive* definition of $f_E(T)$:

$$f_E(T) = \begin{cases} \mathbf{0} & \text{if } T \text{ is empty} \\ \tau_{NN}[L(root), f_E(T^{(1)}), ..., f_E(T^{(k)})] & \end{cases} \qquad (10)$$

where $\mathbf{0}$ is the null vector in $\mathcal{R}^m$, *root* is the root node (or supersource of the tree T), $L(root)$ is the label attached to the root, and $T^{(1)}$, ..., $T^{(k)}$ are the subtrees pointed by *root*. Note that the same definition may be applied to DPAGs once the supersource *s* corresponds to the root of the tree.

Equation 10 comprehensively defines the functionality of the recursive neural network. The recursive definition of $f_E(T)$ determines a systematic visit of the input tree T. It guides the application of τ_{NN} to each node of the structures, from the frontier to the root of the input tree, allowing the neural model to incrementally compute a numerical code for the whole structure. The process corresponds to a postorder traversal of the input tree T that entails the recursive processing of all of the subtrees and finally of the root of T.

Let us consider an example that allows us to grasp the recursive encoding of structures. In Figure 2, we describe the encoding process, visualizing through boxes, from the darkest to the lightest, the progressive processing of the input tree performed by the RecNN. This corresponds to "unfolding" eq 10 through the current input structure.

It is easy to observe that the encoding process is modeled to the morphology of each input compound and the encoding process is adaptive via the free parameters of τ_{NN}.

With respect to the neural realization of τ_{NN} (eqs 8 and 9), we can observe that the connection $\hat{\mathbf{W}}^j$ and the internal state variables $\mathbf{x}^j$ are the machinery able to store information from the current structure and to use it together with the current input (the label of the node). Moreover, because of the learning capabilities, the memory of a recursive neural network is a dynamic memory that is dependent on the task.

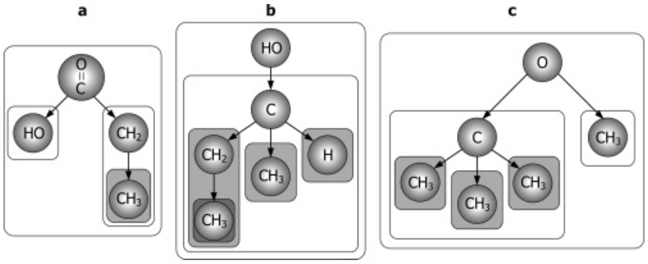


Figure 2. Unfolding the encoding process through structures. Each box includes the subtree progressively encoded by the recursive neural network. The process begins from the darkest box. Molecular graphs for a, propanoic acid; b, 2-butanol; and c, *tert*-butyl methyl ether.

The state-neural units discover adaptive abstract representations of structural data containing the information relevant to the prediction.

To produce the final prediction value, the RecNN model is completed by the output function g, which can be realized by choosing any known mathematical model. In the class of neurocomputing models, g may be obtained using a multilayer network to perform regression or classification tasks. Here, we use a single linear output neuron to realize a regression model:

$$y = g(\mathbf{x}) = \mathbf{A}^t\mathbf{x} + \theta \tag{11}$$

where A $\in \mathscr{R}^m$ and θ is the output threshold.

Training Algorithm. Because, as for standard feed-forward neural networks, the error function is differentiable and the weights of the model define a continuously parametrized space of functions, the learning algorithm of the RecNN can still be based on a gradient descent technique. However, it must be adapted to face the contextual nature of the approximation task, in this case, a map from the set of trees to output values. The learning algorithm must account for all of the sets of computational steps in eq 10; that is, the computation of the gradient must take into consideration not just the current input node but also all input nodes seen in the encoding process by the RecNN. Standard supervised algorithms for feed-forward neural networks have been extended to deal with structures, that is, for RecNN.[48] In the learning algorithm of recursive neural networks, encoding of the structures and the iterative weights updates are interleaved. For RecNNs of the type described so far, we can outline a general learning algorithm as in Chart 1.

There are different ways to realize the recursive neural network.[48] In the present work, we choose to use a constructive approach that allows the training algorithm to progressively add the hidden recursive neurons during the training phase. The model is a recursive extension of cascade-correlation-based algorithms.[50,51] The built neural network has a hidden layer composed of recursive hidden units. The recursive hidden units compute the values of f_E (in $\mathscr{R}^m$) for each input tree. The number of hidden units, that is, the dimension m of the descriptor space, is automatically computed by the training algorithm, thus allowing an adaptive computation of the number and type of numerical descriptors needed for a specific QSPR/QSAR task. Differently from the previous approaches, this implies that no a priori selection or extraction of features or properties is needed in the new scheme for f_E.

Chart 1. Outline of the Learning Algorithm for a Training Set Constituted by Couples (T, d), Where T Is a Tree and d the Corresponding Target Property Value

```
Repeat
  For each example of the training set $(T,d)$
    Encode the input tree $T$: compute $f_E$ through $T$ applying $\tau_{NN}$ for each node of
      $T$ following a post-order traversal of the tree
    Compute the output value for $T$ by $g$, i.e. $g\left(f_E(T)\right)$
    Compute the error evaluating the differences between the target value $d$ and
      $g\left(f_E(T)\right)$
    Compute the gradient of error for each weight of the model accounting for the
      influences of the weight for each step of the encoding process: the error is
      back-propagated unfolding the encoding process through the structure of $T$
  Update the weights $\mathbf{W}$ of $g$ and $\tau_{NN}$ according to the value of the gradient
  computed over the structures of the training set
  Until convergence.
```

A complete description of the RecNN algorithm and a formulation of the learning method and equations can be found in ref 1.

When, as in the present application, a rather low number of training data is available, special care has to be paid to avoid overfitting. Several expedients can be used for this purpose. First of all, a reduced RecNN architecture is built because no connection between hidden units is allowed. Then, the gain of the sigmoids of the hidden units is set to 0.4. Specifically, an incremental strategy (i-strategy)[1] on the number of training epochs can be adopted for each new inserted hidden unit of the RecNN model. Allowing few epochs to the first units, we avoid the increase of the weight values. The advantages of this strategy are already shown in ref 3. The work of Bartlett[52] gives theoretical support for techniques, like the i-strategy, that allow the production of networks with small weights. As a result, we can continue the learning, adding new hidden units in RecNN, without overtraining the model. Anyway, in the present work, we also performed experiments aimed at studying the behavior of the model with different fitting conditions. The results demonstrate that, actually, an early stopping of the training convergence does not improve the general performance.

Representational Issues. As mentioned above, in our approach, the representation of the molecule is directly inferred from the molecular structure alone. For this purpose, we chose to describe the molecule by means of a 2D graph easily obtained from the structure formula. The overall procedure leads, in general, to a loss of information with respect to a 3D representation. However, part of the lost information can be recovered by introducing an order among the atomic clusters individuated as the "groups" constituting the molecule, thus matching, in some sense, the well-known Newmann projections employed to assign the absolute configuration of a chiral molecule.

We have represented the molecular structures in terms of labeled rooted ordered trees. In this light, we decided on a set of rules allowing for the obtainment of a unique structured representation of every molecule of the data set including alkanes, alkenes, alkynes, alcohols, ethers, thiols, thioethers, aldehydes, ketones, carboxylic acids, esters, amines, amides, haloalkanes, nitriles, and nitroalkanes. The adopted rules are summarized in Chart 2.

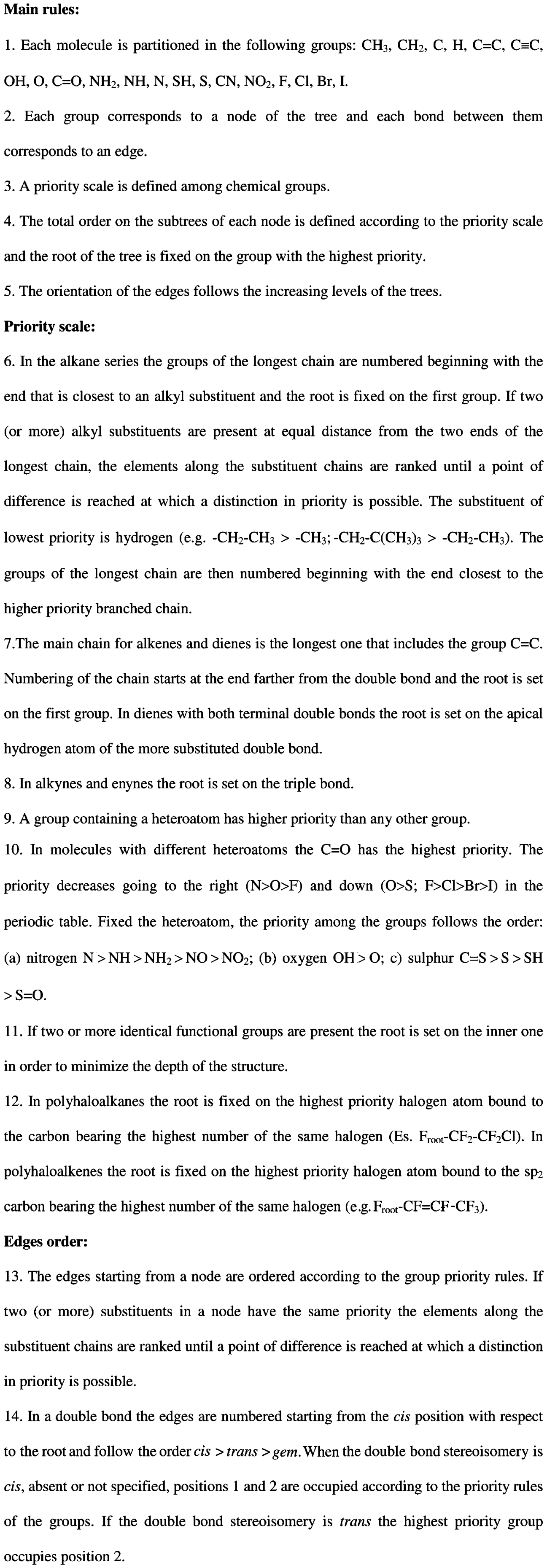
It must be stressed that the above-defined atomic groups coincide only partially with the functional groups identifying the different classes of organic compounds. In particular, the groups COOH, COO−, CONH$_2$, CO−N<, and COX (X = Cl, Br, I) were not defined and were represented as subtrees constituted by two atomic groups (e.g., the group COOH is represented as the composition of the group C═O and the group OH). Moreover, the choice of attributing to the C═O group the highest priority allows for giving a similar representation to ketones, aldheydes, carboxylic acids, esthers, alkanoyl halides, anhydrides, and amides. We chose to divide the CH group into C and H. In such a way, we maintain the same approach in describing the C−H bond independently of the hybridization of the carbon atom. It is worthy to note that, by using the total order of the subtrees, we were able to build up different representations for the cis and trans isomers of alkenes (see Chart 2). Moreover, we can distinguish R and S enantiomers by ordering the edges of the asymmetric carbon according to the priority rules. The second enantiomer is obtained by changing the order of two edges.

A numerical label is associated with each node. The labels discriminate among different groups of atoms and do not contain any physicochemical information. To represent the labels, we use a 1-of-*n* coding scheme for categorical variables. In this way, each label results in a 20-bit vector, with one or a few specific bits turned on (+1) and all the others turned off (0). Sharing bits between different labels allows for representation of the similarity between chemical groups. On the other hand, two orthonormal vectors (i.e., bits turned on in different positions within the vector) represent groups of different chemical natures. In particular, we stated that all of the H groups have the same numerical label; CH$_3$, CH$_2$, and C have similar numerical labels; NH$_2$ and NH have similar numerical labels, but orthonormal to N; OH and O as well as SH and S have orthonormal labels; and F, Cl, Br, and I have similar numerical labels.

The similarity or the orthonormality between groups, assigned according to their chemical features, is the only available chemical information transferred to the RecNN as input data.

In Figure 3 is reported, as an example, the representation of 2-methyl-2 propanol as a chemical tree and the conversion of the tree in the input data file for the RecNN. The input data file contains the dimension of the tree (number of nodes), the value of the target property, and the connection table of the structure. In this table, the first column represents the order number identifying the specific group indicated in column 2; columns 3−5 indicate for each node the presence of a "child" identified by its order number, that is, a pointer to the substructure rooted in the child (−1 means the absence of a child); column 6 reports the number identifying the numerical label associated with the group. In the same figure, the numerical labels, corresponding to the groups present in 2-methyl-2-propanol, are also reported. This representation of the input data is used to recursively read the structure by a recursive neural network as previously shown.

Chart 2. Rules for the Representation of the Molecular Structure



Main rules:

1. Each molecule is partitioned in the following groups: CH$_3$, CH$_2$, C, H, C=C, C≡C, OH, O, C=O, NH$_2$, NH, N, SH, S, CN, NO$_2$, F, Cl, Br, I.

2. Each group corresponds to a node of the tree and each bond between them corresponds to an edge.

3. A priority scale is defined among chemical groups.

4. The total order on the subtrees of each node is defined according to the priority scale and the root of the tree is fixed on the group with the highest priority.

5. The orientation of the edges follows the increasing levels of the trees.

Priority scale:

6. In the alkane series the groups of the longest chain are numbered beginning with the end that is closest to an alkyl substituent and the root is fixed on the first group. If two (or more) alkyl substituents are present at equal distance from the two ends of the longest chain, the elements along the substituent chains are ranked until a point of difference is reached at which a distinction in priority is possible. The substituent of lowest priority is hydrogen (e.g. $-CH_2-CH_3 > -CH_3$; $-CH_2-C(CH_3)_3 > -CH_2-CH_3$). The groups of the longest chain are then numbered beginning with the end closest to the higher priority branched chain.

7.The main chain for alkenes and dienes is the longest one that includes the group C=C. Numbering of the chain starts at the end farther from the double bond and the root is set on the first group. In dienes with both terminal double bonds the root is set on the apical hydrogen atom of the more substituted double bond.

8. In alkynes and enynes the root is set on the triple bond.

9. A group containing a heteroatom has higher priority than any other group.

10. In molecules with different heteroatoms the C=O has the highest priority. The priority decreases going to the right (N>O>F) and down (O>S; F>Cl>Br>I) in the periodic table. Fixed the heteroatom, the priority among the groups follows the order: (a) nitrogen $N > NH > NH_2 > NO > NO_2$; (b) oxygen $OH > O$; c) sulphur $C=S > S > SH > S=O$.

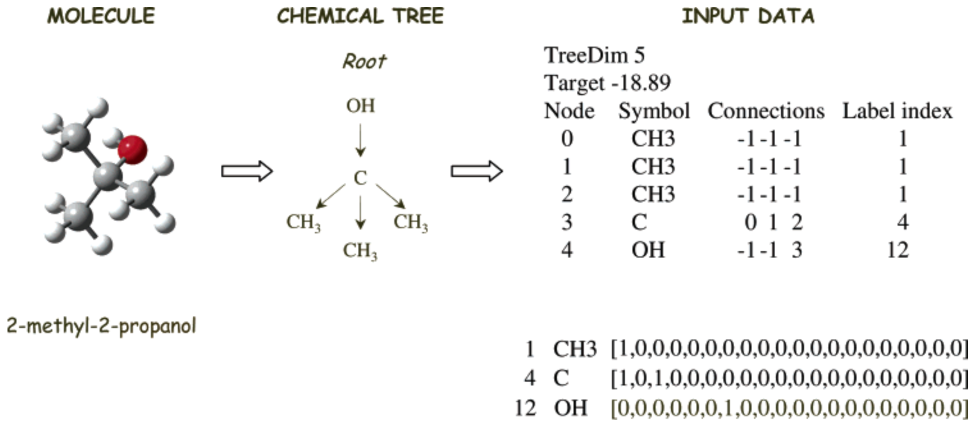
11. If two or more identical functional groups are present the root is set on the inner one in order to minimize the depth of the structure.

12. In polyhaloalkanes the root is fixed on the highest priority halogen atom bound to the carbon bearing the highest number of the same halogen (Es. $F_{root}-CF_2-CF_2Cl$). In polyhaloalkenes the root is fixed on the highest priority halogen atom bound to the sp$_2$ carbon bearing the highest number of the same halogen (e.g. $F_{root}-CF=CF-CF_3$).

Edges order:

13. The edges starting from a node are ordered according to the group priority rules. If two (or more) substituents in a node have the same priority the elements along the substituent chains are ranked until a point of difference is reached at which a distinction in priority is possible.

14. In a double bond the edges are numbered starting from the *cis* position with respect to the root and follow the order *cis* > *trans* > *gem*. When the double bond stereoisomery is *cis*, absent or not specified, positions 1 and 2 are occupied according to the priority rules of the groups. If the double bond stereoisomery is *trans* the highest priority group occupies position 2.



```
TreeDim 5
Target -18.89
Node  Symbol  Connections  Label index
  0    CH3      -1 -1 -1       1
  1    CH3      -1 -1 -1       1
  2    CH3      -1 -1 -1       1
  3    C         0  1  2       4
  4    OH       -1 -1  3      12

 1  CH3  [1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
 4  C    [1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
12  OH   [0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0]
```

Figure 3. Representation of 2-methyl-2-propanol as a chemical tree and input data file containing the dimension of the tree, the target property value, and the connection table of the structure. The numerical labels for CH_3, C, and OH groups are also reported.

RESULTS AND DISCUSSION

Model Evaluation. The above-described RecNN model has been applied to a data set of 179 standard free energies of solvation in water of monofunctional acyclic compounds including alkanes, alkenes, alkynes, alcohols, ethers, thiols, thioethers, aldehydes, ketones, carboxylic acids, esters, amines, amides, haloalkanes, nitriles, and nitroalkanes. The experimental $\Delta_{solv}G°$ values were taken from the data used by Cabani and co-workers.[32] The whole set of molecules was divided into disjointed training and test sets for the learning and validation processes, respectively. The molecules were selected so that the test set was representative of the different molecular sizes, topologies, and functional groups. Three different splitting strategies of the data were used.

In *experiment 1*, the test set (33 compounds) was chosen as it contains the compounds giving the best performance with the group contributions method, GCM, proposed by Cabani and co-workers,[32] when applied to the whole set of compounds. The target values in the training set ranged from -40 to 12 kJ mol^{-1}. A trial using dimensionless target values, normalized in the range 0$-$1, was also run without any significant improvement in the performance. In *experiment 2*, the training set was obtained from the first one by removing three compounds: methane, acetamide, and fluoromethane. Methane was removed because of its peculiar structure. In fact, independently of the way we choose the atomic groups in which to divide the molecule (CH_4 or CH_3 + H or CH_2 + 2H, ...) and the priority rules we adopt, the resulting molecular tree is completely different from the trees of the other alkanes. Acetamide and fluoromethane were removed because they are the only representatives of the respective classes of compounds in the molecules set. Moreover, thiobismethane was moved from the test to the training set given that the S group was scarcely represented. As a consequence of the changes, the target values of the training set in this experiment ranged from -28 to 12 kJ mol^{-1}. In *experiment 3*, fluoromethane was removed from the data set. In addition, we tested the performance of the RecNN in some challenging conditions. To this aim, three disjointed sets of molecules were defined: training, test, and "guess" test sets. The guess test set was constituted by seven compounds (methane, 1-buten-3-yne, 2-propen-1-ol, 2,4-hexadienal, acetamide, chloroethene, and 3-chloro-1-propene) whose molecular features were scarcely represented in the whole data set. More specifically, all of these molecules, except methane, contain two or more atomic groups whose combination is not represented in any molecule of the training set. As regards methane, this compound represents a sort of outbound extrapolation from the alkanes series. We decided to represent the molecule as the combination of CH_3 and H groups. This choice arbitrarily considers one hydrogen atom different from the others but allows us to maintain the same priority rules already fixed for the alkane series. It must be stressed that, for this special guess test set, the predicted values have to be evaluated individually and not statistically. A test set (33 compounds) was randomly chosen for the validation process, while the remaining 138 compounds were used for the learning procedure, their target values still ranging from -28 to 12 kJ mol^{-1}. In this way, we were able to obtain results not influenced by a priori choices and to test in the meantime the sensitivity of the network to different learning conditions. In each experiment, eight trials were carried out. The initial connection weights used in each trial were randomly set. For experiments 1$-$3, learning was stopped when the RecNN inserted 100 hidden units, resulting in a mean training error lower than 0.1 kJ mol^{-1}, this tolerance being below the experimental error on $\Delta_{solv}G°$. A further experiment, *experiment 4*, was also carried out, using the data splitting of experiment 3. In this case, the learning procedure was stopped when the maximum error for any compounds was below 0.45 kJ mol^{-1}, which is close to the standard deviation of the GCM method.[32] The complete list of investigated compounds, the corresponding values of the target property ($\Delta_{solv}G°$), and the mean error for each performed experiment are reported in the Supporting Information (Table S1).

The main statistics computed over all of the experiments are shown in Table 2. Specifically, the mean absolute error, the maximum absolute error, the correlation coefficient, and the standard deviation are reported as obtained by an ensemble averaging method, that is, computing the mean output over the performed trials. The number of hidden units of each experiment is also reported.

We can observe that, independently of the learning conditions, the training values are reproduced within the experimental error (0.1 kJ mol^{-1}), while in the test set, the standard deviations are about 0.7 kJ mol^{-1}.The differences among the statistical parameters in the four experiments are quite low, and only in the fourth one do the standard errors

Table 2. Number of Hidden Units, *U*; Mean and Maximum Absolute Error; Correlation Coefficient, *R*; and Standard Deviation, *S*, of the Different Experiments[a]

		training set				test set			
		absolute error				absolute error			
expt.	*U*	mean[b]	max.	*R*[c]	*S*	mean[b]	max.	*R*[c]	*S*
1	100	0.05	0.22	0.999 98	0.07	0.40	1.19	0.9987	0.50
2	100	0.06	0.28	0.999 96	0.09	0.53	2.63	0.9972	0.77
3	100	0.06	0.33	0.999 98	0.08	0.46	2.12	0.9985	0.68
4	72[d]	0.09	0.43	0.999 95	0.12	0.49	2.39	0.9982	0.73

[a] All of the statistical parameters, except *R*, are expressed in kJ mol^{-1}. [b] Mean absolute error defined as the mean of the absolute residuals. [c] Linear correlation coefficient between the experimental and calculated values. [d] Average of the number of hidden units over the eight trials.

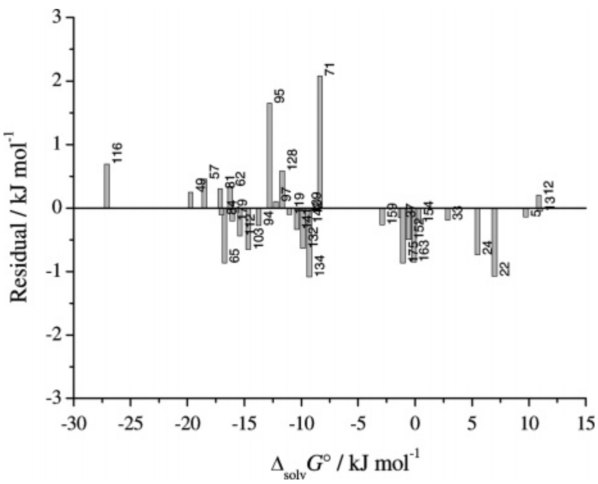


Figure 4. Plot of the residuals for the test set in experiment 3. The compounds are individuated by their order number as indicated in Table S1 of the Supporting Information.

on the training set significantly increase. Indeed, this is an interesting result because, as mentioned above, in the case of experiment 4, the training procedure was stopped when a higher error threshold was attained and a corresponding lower number of hidden units was involved in the calculation. Because an early stopping of training does not improve the general performance of the model, we can take this result as a direct experimental proof that overtraining conditions have been avoided in all of the experiments, the applied i-strategy was effective, and the good fitting of the data achieved in our experiments is not at the expense of the predictive accuracy. As a further remark, we can say that the different choice of the test set molecules does not significantly affect the results.

It can be noticed that larger errors are frequently found for molecules with five carbon atoms or less in the skeleton. In particular, C5 compounds are usually the most critical in all of the classes. Though the adaptive process is probably less effective when a short, unbranched tree is processed by RecNN, at the moment, no univocal simple explanation can be given for this trend. Figure 4 reports the plot of the residuals obtained by RecNN for the test set in experiment 3. It is worthy to note that $\Delta_{solv}G°$ values of carboxylic acids, esters, and aldheydes are reliably predicted even if the COOH, COO−, and CHO groups are not defined as independent groups. In fact, in the representation of the molecules we used, COOH and COO are considered as the combination of CO and OH groups and CO and O groups, respectively. In the same way, the aldehydic group CHO is represented as a subtree constituted by CO and H groups. On the other hand, the analysis of the residuals shows that these choices do not reduce the reliability of the $\Delta_{solv}G°$ prediction of the molecules which contain only one of these groups. We consider especially profitable the choice of dividing the CH group into a C and an H atom, despite the hybridization of the carbon atom. This allowed us to use the same label for the H atom in the CH group of alkanes, alkenes, and alkynes as well as in the CHO group of aldheydes or methanoic acid.

Table 3. Experimental $\Delta_{solv}G°$ Values and Mean Errors on the Guess Test Set Obtained in Experiment 3[a]

molecule	$\Delta_{solv}G°$	δ RecNN
methane	8.37	1.15
1-buten-3-yne	0.17	−5.02
2-propen-1-ol	−21.06	−2.01
2,4-hexadienal	−19.39	5.69
acetamide	−40.63	15.69
chloroethene	−2.48	−0.60
3-chloro-1-propene	−2.4	−1.46

[a] All values in kJ mol^{-1}.

In Table 3, the experimental Gibbs energies of solvation of the molecules selected for the guess test set are reported together with the mean errors, δ, calculated as the difference between the predicted and the experimental values. The average error is reported as computed over the eight trials of experiment 3.

It may be observed that the values predicted for methane, 2-propen-1-ol, and chloroalkenes are quite good. The result obtained for methane is particularly meaningful, and it confirms the effectiveness of the representation adopted for this compound. As regards the high mean error obtained for acetamide, we have to stress that the experimental $\Delta_{solv}G°$ of −40 kJ mol^{-1} of this compound is out of the experience of the neural model. In fact, in the training set of experiment 3, the $\Delta_{solv}G°$ data range from −28 to 12 kJ mol^{-1}. RecNN correctly assigns to acetamide a predicted value close to the negative limit of the target data range. A rather high error is also found for 1-buten-3-yne and 2,4-hexadienal. In both cases, the RecNN is not trained at all over these classes of conjugated compounds. For the first one, the neural network can only infer the target property by averaging between the behavior of alkenes and alkynes, thus predicting a more negative value. In fact, analyzing the experimental data, we find that the substitution of an ethyl group within an alkyl chain with either a $-CH{=}CH_2$ or a $-C{\equiv}CH$ group produces a large decrease in $\Delta_{solv}G°$. For 2,4-hexadienal, the RecNN finds a balance between the behaviors of α,β-unsaturated aldehydes and conjugated dienes. The predicted value is lower in magnitude with respect to the experimental one. As a further remark, we may also observe that only a few molecules among those included in the training set have highly negative $\Delta_{solv}G°$ values so that the neural network is scarcely trained within this range of target properties. This fact probably concurs to the unsatisfactory prediction of the $\Delta_{solv}G°$ value of 2,4-hexadienal.

Internal Representations and Domain Knowledge. One of the most appealing issues of this RecNN application is understanding whether the proposed model is able to capture

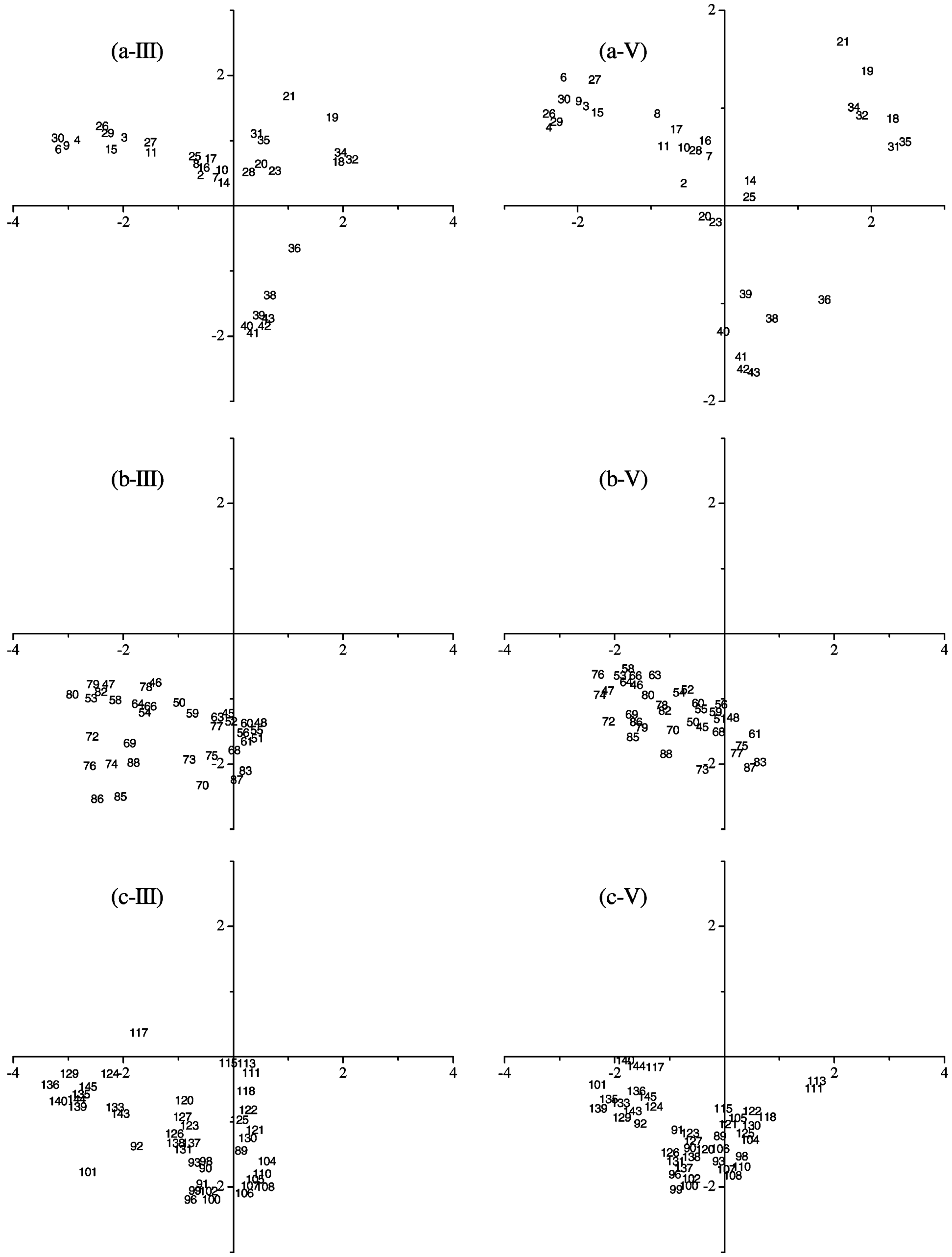


Figure 5. Plots of the two principal components (PC1 abscissa and PC2 ordinate) of training compounds used in experiment 3 derived from trials III and V. (a) alkanes (2−17), alkenes (18−35), and alkynes (36−43); (b) alcohols (45−66), ethers (68−76), and amines (77−88); (c) ketones (89−102), aldehydes (104−113), carboxylic acids (115−117), and esters (118−145). Order number of the compounds as in Table S1 of the Supporting Information.

significant domain knowledge from the training data. To do this, we can investigate the internal representations, that is, the output of hidden units (*X*) computed by the neural network trained with the selected set of molecules. These outputs represent the encoding values generated for each compound or molecular fragment. Therefore, the analysis of internal representations could in principle be done at any level of the molecular tree. However, because of the quite high number of classes of chemical compounds and the complexity and variety of sampled molecular structures, we do prefer to analyze the internal representations only at the root level, that is, by considering the molecule as a whole. Because the number of hidden units is high and the dimension of the representational space is correspondingly large, we performed a principal component analysis (PCA) of the internal representations and studied 2D plots of the first two principal components. These plots show in a rather direct way the relative distance and position of the internal representation, enabling us to infer significant information about the knowledge learned by the neural network from the training data. As the results of the RecNN application are nearly independent of the learning conditions, we applied the PCA analysis only to experiment 3. Each trial of experiment 3 was then separately analyzed by PCA, and the plots of the first two principal components from the trials III and V are reported in Figure 5, as an example. Because of the large number of represented molecules, we have split the set of compounds into three separate plots according to their basic chemical nature, namely, hydrocarbons (alkanes, alkenes, and alkynes), polar compounds (alcohols, ethers, and amines), and carbonyl compounds (aldehydes, ketones, carboxylic acids, and esters). The classes of compounds scarcely represented in the data set have been neglected.

As we can see, the molecules are clustered in a well-defined area according to their own class. In particular, it can be noticed that alkanes, alkenes, and alkynes occupy the second, first, and fourth quadrants, respectively, whereas polar and carbonyl compounds approximately lay in the third one. In other words, molecules whose molecular trees have different structures and different roots lay in the same region. This means that the observed groupings are not due to the fact that the internal representation simply retains memory of the molecular graph. Indeed, these groupings represent a knowledge of chemical features that cannot be directly inferred from the molecular graph but only by the association of the molecular structure to the target property. Definitely, they represent what the recursive neural network did learn about the chemical features of the molecules. In this frame, we can make some observations by analyzing the distribution of the points in each class. As regards alkenes, we can observe that, though generally laying in the first quadrant, some of them are scattered in the alkanes region. More specifically, this happens for long chains or alkyl-substituted alkenes, that is, when the alkyl portion prevails in the molecule. On the other hand, the internal representations proved to be very effective in discriminating among isomeric compounds. For instance, 2-methylpentane (10) and 3-methylpentane (11), as well as 2-methyl-1-pentene (27) and 3-methyl-1-pentene (28), lay significantly apart from each other in the plots. In particular, the case of the isomeric alkanes is meaningful because they cannot be discriminated by a standard group contributions method. On the contrary, though laying in the alkenes region, dienes show a quite unexpected behavior: 1,3-butadiene (31) and 2,3-dimethyl-1,3-butadiene (35) change significantly their positions in the plots of trials III and V but still remain very close to each other.

The most appealing features emerging from the PCA analysis probably concern the behavior of polar and carbonyl compounds. As mentioned above, all of them are grouped in a big cloud laying in the third quadrant (and marginally in the fourth one). However, when the individual points are taken into consideration, some very interesting results emerge. For instance, it can be noticed that methanol (45) and methylamine (77), ethanol (46) and ethylamine (78), and 1-propanol (47) and 1-propylamine (79) lay very close to each other, especially in the plot of trial III. This should mean that, in the representational space, an alkanol is much more similar to a primary amine with the same number of carbon atoms than to the corresponding superior homologous one. This is a largely unexpected result because both of the trees of the two molecules have a completely different root and, the target property is also different. Moreover, alcohols and primary amines are clustered close to the upper boundary of the cloud, while ethers and tertiary amines lay significantly below it. This seems to suggest that the accessibility of the polar group by the solvent and its ability to act as a hydrogen-bond donor or acceptor is responsible for the distribution of polar molecules in the representational space. This behavior seems to be confirmed also by the plot of carbonyl compounds, where carboxylic acids and aldehydes lay close to the upper boundary while ketones and esters are located below. It has to be stressed that this kind of chemical knowledge abstracted by the RecNN does not appear trivially decoupled into single effects. On the contrary, the model combines these features, developing a sort of "smooth rule", reflected by the spread of the points in the clusters, globally accounting for the complexity of the stereoelectronic properties of molecules.

CONCLUSIONS

The proposed RecNN model allows for a completely novel approach to QSPR analysis. We consider the results obtained until now through this approach as very promising.

The original representation of the molecules developed in this work takes into account both the occurrences of specific atom/groups in the compound and the topological relationships expressed by the structure and demonstrates to be very flexible and effective in dealing with the generality of structures.

Our model shows a better descriptive and predictive ability than the standard QSPR approaches, using multilinear regression analysis, and it matches the performances obtained by neural-network-based methods. On the other hand, we believe that the main result of our approach lies in the methodological novelty in the handling of the molecular structure. In fact, while the most advanced literature methods are highly tuned methods exploiting the known descriptors and the background knowledge in the field, our model directly takes a variable-size hierarchical labeled structure as input. Both the encoding and the mapping functions are simultaneously and automatically learned by a process of training from examples. In this way, the model overcomes the limits of a vectorial representation of the molecules and avoids the need of an a priori selection of the molecular descriptors. However, it must be observed that the literature models proposed for the prediction of $\Delta_{\mathrm{solv}}G°$ have been applied to a wider data set representative of the generality of chemical structure, while our model has been tested on a data set containing only monofunctional compounds. For this reason, we plan to extend this study to sets of compounds spanning over a widespread survey of chemical structures and functionalities and including polyfunctional, cyclic, aromatic, and heteroaromatic compounds. Conversely, we believe that the predicting ability of our RecNN model should be improved by increasing the learning basis. In fact, the worst results are obtained in the case of molecules whose functional groups and structures are scarcely represented in the training set. The principal component analysis of the internal representation computed by the RecNN for the compounds considered in this work was able to give a glimpse of the molecular features extracted by the RecNN as the most significant for the $\Delta_{\mathrm{solv}}G°$ prediction. It is worthy to note that the chemical knowledge taken out by the model globally accounts for the complexity of the stereoelectronic properties of molecules.

On the basis of the quantitative and qualitative results, we can conclude that the proposed approach introduces a relevant advancement in predicting physical–chemical properties of compounds from their molecular structure.

ACKNOWLEDGMENT

This work has been financially supported by Ministero dell'Istruzione, dell'Università e della Ricerca, Roma (PRIN

2005 MIUR). The authors thank Prof. Benedetta Mennucci for useful discussions.

Supporting Information Available: For each examined compound, the value of $\Delta_{\rm solv}G°$ and the values of the mean error, δ, for experiments 1−4 are reported. δ is defined as the difference between calculated and experimental $\Delta_{\rm solv}G°$ averaged over the performed trials. This information is available free of charge via the Internet at http://pubs.acs.org.

REFERENCES AND NOTES

(1) Bianucci, A. M.; Micheli, A.; Sperduti, A.; Starita, A. Application of Cascade Correlation Networks for Structures to Chemistry. *Appl. Int. J.* **2000**, *12*, 117−147.

(2) Micheli, A. Recursive Processing of Structured Domains in Machine Learning. Ph.D. Thesis TD-13/03, Dipartimento di Informatica, University of Pisa, Pisa, Italy, 2003.

(3) Micheli, A.; Sperduti, A.; Starita, A.; Bianucci, A. M. Analysis of the Internal Representations Developed by Neural Networks for Structures Applied to Quantitative Structure−Activity Relationship Studies of Benzodiazepines. *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 202−218.

(4) Micheli, A.; Sperduti, A.; Starita, A.; Bianucci, A. M. A Novel Approach to QSPR/QSAR Based on Neural Networks for Structures. In *Soft Computing Approaches in Chemistry*; Sztandera, L. M., Cartwright, H. M., Eds.; Springer-Verlag: Heidelberg, Germany, 2003; pp 265−296.

(5) Ben-Naim, A. *Solvation Thermodynamics*; Plenum Press: New York, 1987.

(6) Jorgensen, W. L.; Tirado-Rives, J. Free Energies of Hydration for Organic Molecules from Monte Carlo Simulations *Persp. Drug Discovery Des.* **1995**, *3*, 123−138.

(7) Tomasi, J.; Persico, M. Molecular Interaction in Solution: An Overview of Methods Based on Continuous Distributions of the Solvent. *Chem. Rev.* **1994**, *94*, 2027−2094.

(8) Cramer, C. J.; Truhlar D. G. Continuum Solvation Models: Classical and Quantum Mechanical Implementations. In *Reviews in Computational Chemistry*; Lipkowitz, K. B., Boyd, D. B., Eds.; VCH Publishers: New York, 1995; Vol. 6.

(9) Hawkins, G. D.; Liotard, D. A.; Cramer, C. J.; Truhlar, D. G. Omnisol: Fast Prediction of Free Energies of Solvation and Partition Coefficients. *J. Org. Chem.* **1998**, *63*, 4305−4313.

(10) Tomasi, J.; Mennucci, B.; Cammi, R. Quantum Mechanical Continuum Solvation Models. *Chem. Rev.* **2005**, *105*, 2999−3093.

(11) Miertus, S.; Scrocco, E.; Tomasi, J. Electrostatic Interaction of a Solute with a Continuum. A Direct Utilization of ab initio Molecular Potentials for the Prevision of Solvent Effects. *J. Chem. Phys.* **1981**, *55*, 117−129.

(12) Barone, V.; Cossi, M.; Tomasi, J. A New Definition of Cavities for the Computation of Solvation Free Energies by the Polarizable Continuum Model. *J. Chem. Phys.* **1997**, *107* (8), 3210−3221.

(13) Klamt, A.; Jonas, V.; Bürger, T.; Lohrenz J. C. W. Refinement and Parametrization of COSMO−RS. *J. Phys. Chem. A* **1998**, *102*, 5074−5085.

(14) Kelly, C. P.; Cramer, C. J.; Truhlar, D. G. SM6: A Density Functional Theory Continuum Solvation Model for Calculating Aqueous Solvation Free Energies of Neutrals, Ions, and Solute-Water Clusters. *J. Chem. Theory Comput.* **2005**, *1*, 1133−1152.

(15) Torrens, F. Universal Organic Solvent−Water Partition Coefficient Model. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 236−240.

(16) Pascal, P. Program SCAP, Université Henry Poincaré, Nancy Cedex, France, 1991.

(17) Metropolis, N.; Rosenbluth, A. W.; Rosenbluth, M. N.; Teller, A. H.; Teller, E. Equation of State Calculations by Fast Computing Machines. *J. Chem. Phys.* **1953**, *21*, 1087−1092.

(18) Alder, B. J.; Wainwright, T. E. Phase Transition for a Hard-Sphere System. *J. Chem. Phys.* **1957**, *27*, 1208.

(19) Rahman, A.; Stillinger, F. M. Molecular Dynamics Study of Liquid Water. *J. Chem. Phys.* **1971**, *52*, 3336−3359.

(20) Duffy, E. M.; Jorgensen, W. L. Prediction of Properties from Simulations: Free Energies of Solvation in Hexadecane, Octanol, and Water. *J. Am. Chem. Soc.* **2000**, *122*, 2878−2888.

(21) Murray, J. S.; Abu-Awwad, F.; Politzer, P. Prediction of Aqueous Solvation Free Energies from Properties of Solute Molecular Surface Electrostatic Potentials. *J. Phys. Chem. A* **1999**, *103*, 1853−1856.

(22) Katritzky, A. R.; Oliferenko, A. A.; Oliferenko, P. V.; Petrukhin, R.; Tatham, D. B.; Uko Maran, U.; Lomaka, A. L.; Acree, W. E., Jr. A General Treatment of Solubility. 1. The QSPR Correlation of Solvation Free Energies of Single Solutes in Series of Solvents. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 1794−1805.

(23) Katritzky, A. R.; Oliferenko, A. A.; Oliferenko, P. V.; Petrukhin, R.; Tatham, D. B.; Uko Maran, U.; Lomaka, A. L.; Acree, W. E., Jr. A General Treatment of Solubility. 2. The QSPR Correlation of Solvation Free Energies of Single Solutes in Series of Solvents. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 1806−1814.

(24) Yaffe, D.; Cohen, Y.; Espinosa, G.; Arenas, A.; Giralt, F. A Fuzzy ARTMAP−Based Quantitative Structure−Property Relationship (QSPR) for the Henry's Law Constant of Organic Compounds. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 85−112.

(25) Abboud, J. L.; Kamlet, M. J.; Taft, R. W. Regarding a Generalized Scale of Solvent Polarities. *J. Am. Chem. Soc.* **1977**, *99*, 8325−8327.

(26) Taft, R. W.; Abraham, M. H.; Famini, G. R.; Doherty, R. M.; Abboud, J. L. M.; Kamlet, M. J. Solubility Properties in Polymers and Biological Media. 5. An Analysis of the Physicochemical Properties which Influence Octanol−Water Partition Coefficients of Aliphatic and Aromatic Solutes. *J. Pharm. Sci.* **1985**, *74*, 807−814.

(27) Abraham, M. H.; Andonian-Haftvan, J.; Whiting, G. S.; Leo, A.; Taft, R. S. Hydrogen Bonding. Part 34. The Factors that Influence The Solubility of Gases and Vapours in Water at 298 K, and a New Method for Its Determination. *J. Chem. Soc., Perkin. Trans 2.* **1994**, 1777−1791.

(28) Abraham, M. H.; Zissimos, A. M.; Acree, W. E., Jr. Partition of Solutes from the Gas Phase and from Water to Wet and Dry di-*n*-Butyl Ether: A Linear Free Energy Relationship Analysis. *Phys. Chem. Chem. Phys.* **2001**, *3*, 3732−3736.

(29) Abraham, M. H.; Platts, J. A. Hydrogen Bond Structural Group Constants. *J. Org. Chem.* **2001**, *66*, 3484−3491.

(30) Hine, J.; Mookerjee, P. K. The Intrinsic Hydrophilic Character of Organic Compounds. Correlations in Terms of Structural Contributions. *J. Org. Chem.* **1975**, *40*, 292−298.

(31) Meylan, W. M.; Howard, P. H. Bond Contribution Method for Estimating Henry's Law Constants. *Environ. Toxicol. Chem.* **1991**, *10*, 1283−1293.

(32) Cabani, S.; Gianni, P.; Mollica, V.; Lepori, L. Group Contribution to the Thermodynamic properties of Non-Ionic Organic Solutes in Dilute Aqueous Solution. *J. Solution Chem.* **1981**, *10*, 563.

(33) Viswanadhan, V. N.; Ghose, A. K.; Singh, U. C.; Wendoloski, J. J. Prediction of Solvation Free Energies of Small Organic Molecules: Additive-Constitutive Models Based on Molecular Fingerprints and Atomic Constants. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 405−412

(34) Eisenberg, D.; Malachlan, A. D. Solvation Energy in Protein Folding and Binding. *Nature (London)* **1986**, *319*, 199−203.

(35) Hou, T.; Qiao, X.; Zhang, W.; Xu, X. Empirical Aqueous Solvation Models Based on Accessible Surface Areas with Implicit Electrostatics. *J. Phys. Chem. B* **2002**, *106*, 11295−11304.

(36) Nirmalakhandan, N. N.; Speece, R. E. QSAR Model for Predicting Henry's Constant. *Environ. Sci. Technol.* **1988**, *22* (11), 1349−1361.

(37) Nirmalakhandan, N.; Brennan, R. A.; Speece, R. E. Predicting Henry's Law Constant and the Effect of Temperature on Henry's Law Constant. *Water Res.* **1997**, *31* (6), 1471−1481.

(38) Brennan, R. A.; Nirmalakhandan, N.; Speece, R. E. Comparison of Predictive Methods for Henrys Law Coefficients of Organic Chemicals. *Water Res.* **1998**, *32* (6), 1901−1911

(39) Russell, C. J.; Dixon, S. L.; Jurs, P. C. Computer Assisted Study of the Relationship between Molecular Structure and Henry's Law Constant. *Anal. Chem.* **1992**, *64*, 1350−1355.

(40) Todeschini, R.; Consonni, V.; Pavan, M. MobyDigs: Software for Regression and Classification Models by Genetic Algorithms. In *Nature-Inspired Methods in Chemometrics: Genetic Algorithms and Artificial Neural Networks*; Leardi, R., Ed.; Elsevier: Amsterdam, 2003.

(41) Mitchell, B. E.; Jurs, P. C. Prediction of Aqueous Solubility of Organic Compounds from Molecular Structure. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 489−496.

(42) Katritzky, A. R.; Lobanov, V.; Karelson, M. *CODESSA Reference Manual*, version 2.0; University of Florida: Gainesville, FL, 1996. http://www.codessa-pro.com (accessed Feb 2006).

(43) Lučić, B.; Nadramija, D.; Bašic, I.; Trinajstić, N. Toward Generating Simpler QSAR Models: Nonlinear Multivariate Regression versus Several Neural Network Ensembles and Some Related Methods. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 1094−1102.

(44) Katritzky, A. R.; Mu, L.; Karelson, M. A QSPR Study of the Solubility of Gases and Vapors in Water. *J. Chem. Inf. Comput. Sci*. **1996**, *36*, 1162−1168.

(45) Katritzky, A. R.; Sild, S.; Lobanov, V.; Karelson, M. Quantitative Structure−Property Relationship (QSPR) Correlation of Glass Transition Temperatures of High Molecular Weight Polymers. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 300−304.

(46) Katritzky, A. R.; Lomaka, A.; Petrukhin, R.; Jain, R.; Karelson, M.; Visser, A. E.; Rogers, R. D. QSPR Correlation of the Melting Point for Pyridinium Bromides, Potential Ionic Liquids. *J. Chem. Inf. Comput. Sci.* **2002**, *42*, 71−74.

(47) English, N. J.; Carroll, D. G. Prediction of Henry's Law Constants by a Quantitative Structure Property Relationship and Neural Networks. *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 1150−1161.

(48) Sperduti, A.; Starita, A. Supervised Neural Networks for the Classification of Structures. *IEEE Trans. Neural Networks* **1997**, *8*, 714−735.

(49) Frasconi, P.; Gori, M.; Sperduti, A. A General Framework for Adaptive Processing of Data Structures. *IEEE Trans. Neural Networks* **1998**, *9*, 768−786.

(50) Fahlman, S. E.; Lebiere, C. The Cascade-Correlation Learning Architecture. In *Advances in Neural Information Processing Systems 2*; Touretzky, D. S., Ed.; Morgan Kaufmann Publishers: San Mateo, CA, 1990; pp 524−532.

(51) Fahlman, S. E. The Recurrent Cascade Correlation Learning Architecture. In *Advances in Neural Information Processing Systems 3*; Lippmann, R. P., Moody, J. E., Touretzky, D. S., Eds.; Morgan Kaufmann Publishers: San Mateo, CA, 1991; pp 190−196.

(52) Bartlett, P. L. The Sample Complexity of Pattern Classification with Neural Networks: The Size of the Weights Is More Important than the Size of the Network. *IEEE Trans. Inf. Theory* **1998**, *44* (2), 525−536.

CI060104E