Pairwise Interaction Field Neural Networks For Drug Discovery

Alessandro Lusci
alessandro.lusci@ucdconnect.ie
Complex and Adaptive Systems Laboratory and
School of Computer Science and Informatics,
University College Dublin, Ireland

Gianluca Pollastri
gianluca.pollastri@ucd.ie
Complex and Adaptive Systems Laboratory and
School of Computer Science and Informatics,
University College Dublin, Ireland

University College Dublin, Technical Report UCD-CSI-2013-02, June 2013

Abstract

Automatically mapping small, drug-like molecules into their biological activity is an open problem in chemioinformatics. Numerous approaches to solve the problem have been attempted, which typically rely on different machine learning tools and, critically, depend on the how a molecule is represented (be it as a one-dimensional string, a two-dimensional graph, its three-dimensional structure, or a feature vector of some kind). In fact arguably the most critical bottleneck in the process is how to encode the molecule in a way that is both informative and can be dealt with by the machine learning algorithms downstream.

Recently we have introduced an algorithm which entirely does away with this complex, error-prone and time-consuming encoding step by automatically finding an optimal code for a molecule represented as a two-dimensional graph. In this report we introduce a model which we have recently developed (Neural Network Pairwise Interaction Fields) to extend this same approach to molecules represented as their three-dimensional structures. We benchmark the algorithm on a number of public data sets. While our tests confirm that three-dimensional representations are generally less informative than two-dimensional ones (possibly because the former are generally the result of a prediction process, and as such contain noise), the algorithm we introduce compares well with the state of the art in 3D-based prediction, in spite of not requiring any prior knowledge about the domain, or prior encoding of the molecule.

Background

Over the last few decades numerous methods have been developed to perform virtual screening of chemical compounds. Most of these methods belong to the broad category of QSAR (Quantitative Structure-Activity Relationship). The aim of QSAR is to find an appropriate function $F()$, which, given a structured representation of a molecule, predicts its biological activity [25]. QSAR's most general form is:

$$Activity = F(structure) \tag{1}$$

The definition of function $F()$ is a complex task which can be factorized into two sub-problems: the *encoding problem* and the *mapping problem*. The former refers to the task of mapping a molecule, which is naturally described as an undirected graph representing its chemical structure, into an array of features. This step is necessary in order to obtain a representation which is suitable for standard regression/classification tools like Artificial Neural Networks (ANN) or Support Vector Machines (SVM). The latter consists in mapping the array of features into the property of interest and, as mentioned, is generally a regression or a classification task, which may be tackled by one of numerous algorithmic tools that are available. According to this view, $F()$ can be decomposed as follows [25]:

$$F() = g(t()) \tag{2}$$

where $t()$ is the encoding function and $g()$ is the mapping function. The way $t()$ is defined is rather open-ended and ultimately one could argue that the essence of the problem is precisely finding $t()$ or, equivalently, that once an informative $t()$ is found for the problem at hand, the following step is trivial. In most cases $t()$ is hand-crafted and requires the intervention of experts. If this is the case, finding $t()$ is usually time consuming and, given that even experts may fail, or overlook, may lead to the loss of important information to predict the desired target. In [8, 12] and [9] a similar approach is followed to predict acqueous solubility by a Multi Layer Perceptron (MLP) and SVM, respectively. In [14] a large set of molecular features, including physical and graphical properties, is compressed by Principal Component Analysis (PCA) to be the input to an ANN, with the aim of predicting melting points. In [6] numeric codes for alkanes are applied to provide an input for an MLP in order to predict melting points and in [5] a set of 2D and 3D molecular descriptors for each molecule is calculated, to predict melting points using a method based on partial least squares Projection to Latent Structures (PLS). Among all current state-of-the-art automated methods (i.e., where the function $t()$ is defined by a fully automated computational process), one of special interest is represented by N-Dimensional Kernels as described in Azencott et al.[2]. In particular, when the number of examples in the training set is large enough (greater than 1000), 2D spectral kernels proved to yield robust results, generally better than 3D kernels.
Another interesting class of methods is represented by the UG-RNN models developed by Lusci et al.[15]. The latter proved to match and in some cases outperform the generalization capability of state of the art Kernels. Unlike

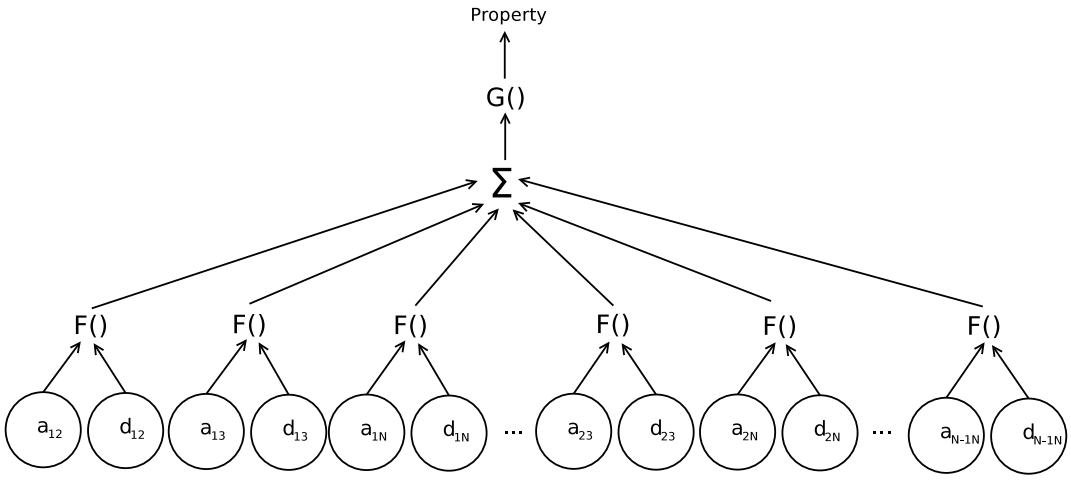

Figure 1: NN-PIF Model of a molecular graph containing N nodes

other methods, UG-RNN is based on a training algorithm where the extraction of molecular features is property driven, i.e. the input label does not consist in a set of pre-computed features.
Here we apply a method based on the algorithm by Martin et al.[17]. The method is called Neural Network Pairwise Interaction Fields. NN-PIF is characterized by a training process where the extraction of molecular features is property driven but, unlike UG-RNN, it relies on 3D molecular graphs. We tested the method on three different standard benchmarks and compared its results with those obtained by a number of state-of-the-art approaches.

Methods

The proposed method is based on the NN-PIF algorithm developed by Martin et al.[17] for protein model quality assessment and ab initio protein folding, and part of a suite of machine learning methods for structured data which we have developed to deal with the prediction of protein structure and function [3, 22, 4, 23, 20, 27, 19, 28, 21, 16, 26, 15, 18]. The input is the 3D graph of a molecule. The output is a property which is assumed to be predictable from the 3D graph. The idea underpinning the algorithm is based on the observation that: each node (i.e. atom) in the graph interacts with its neighbours; the type of interaction depends on the atom types and atomic distances (including where covalent bonds are present). Here we map the interaction between each couple of atoms i and j to a hidden state $X_{i,j}$, through a function $F()$ that takes the atom labels a_i and a_j and the distance between atoms $d_{i,j}$ as input.

$$X_{i,j} = F(a_i, a_j, d_{i,j}) \tag{3}$$

Table 1: Architecture of 10 encoding neural networks N^F and output neural networks N^G

NeuralNetwork	N^F Hidden Units	N^F Output Units	N^G Hidden Units
Model_1	10	3	7
Model_2	10	4	7
Model_3	10	5	7
Model_4	10	6	7
Model_5	10	7	7
Model_6	10	8	7
Model_7	10	9	7
Model_8	10	10	7
Model_9	10	11	7
Model_10	10	12	7

The function $F()$ is implemented by a feed-forward neural network N^F with a single hidden layer with hyperbolic tangent outputs. The hidden states are then combined together for each interaction, yielding a hidden vector Y for the whole molecule:

$$Y = K \sum_{i \neq j} X_{ij} \tag{4}$$

Thus Y is a feature vector encoding the properties of the whole 3D graph (see 1). The hidden vector Y is then mapped into a single output, which represents a single property for the whole structure:

$$O = G(Y) \tag{5}$$

We implement $G()$ as a feed-forward neural network N^G with one hidden layer and a sigmoidal (hyperbolic tangent) output. The functions $F()$ and $G()$ are assumed to be stationary, hence the same network N^F is replicated for all the interactions, and the same network N^G is replicated for all conformations. The overall NN-PIF architecture is trained by gradient descent. The error used is the squared difference between the network output and the desired property. The gradient can be easily computed in closed form, via a version the back-propagation algorithm, as the overall graph does not contain cycles.

NNPIF configuration and training

As mentioned, both N^F and N^G are modelled by neural networks, both of which contain one hidden layer. All neurons use a sigmoid transfer function ($tanh$) and weights are randomly initialized. In order to reduce the residual generalization error[11], we use an ensemble of 10 models with different numbers of hidden units as described in 1.

We trained our NN-PIF models for 30000 epochs with a fixed learning rate $\eta = 0.001$. In order to speed up the learning process[24], we include a momentum term in the training routine. The resulting weights update rule at step e is:

$$\Delta w(e) = \Delta w(e) - m\Delta w(e-1) \tag{6}$$

We chose $m = 0.9$. The outputs of the best 10 networks, selected by their Root Mean Square Error (RMSE) on the validation set, are averaged as an ensemble to compute the prediction on the test set, on each fold of the 10-fold cross validation procedure.

Data

To train and test the NN-PIF we use three publicly available benchmark datasets widely used in the solubility prediction literature. The following datasets do not include 3D coordinates, therefore we used Marvin Beans[1] to predict them. As we discuss later, this is a potential source of noise.

Small Delaney Dataset

This dataset[8] originally contained 2874 molecules together with their measured aqueous solubility ($logmol/L$ at 25 °C). This dataset is particularly interesting because it can be used as a benchmark for comparisons against the GSE method[13]. As described in Delaney[8], the GSE was obtained from a set of molecules similar to the ones contained in the "Small" Delaney Dataset. Furthermore, various kernel methods[2] have also been trained and tested on this dataset with better results than GSE.

Huuskonen

This dataset contains 1026 organic molecules selected by Jarmo Huuskonen[12] from the AQUASOL dATAbASE[29] and the PHYSPROP Database[7]. Molecules are listed together with their aqueous solubility values, expressed in $logmol/L$ at 20-25°. For instance, Frohlich et al.[10] report a squared correlation coefficient of 0.90 for an 8-fold cross-validation, using support vector machines with a RBF (Radial Basis Function) kernel.

Karthikeyan

The dataset consists of 4173 compounds annotated with melting points in degrees Celsius and a wide range of additional properties[14]. In our tests we limit the molecular target to the melting point. The latter is a fundamental physiochemical property of a molecule that depends on both single-molecule properties and intermolecular interactions due to packing in the solid state. Karthikeyan[14] found that models based on 2D descriptors contain more relevant information than 3D descriptors.

Table 2: Prediction performances and standard deviations using 10-fold cross validation on the Small Delaney Dataset (1144 molecules)

Models	R^2	std R^2	RMSE	std RMSE	AAE	std AAE
NN-PIF	0.89	0.03	0.70	0.09	0.53	0.05
UG-RNN[15]	**0.92**	0.02	**0.58**	0.07	**0.43**	0.04
UG-RNN-CR[15]	0.86	0.03	0.79	0.09	0.57	0.06
UG-RNN+LogP[15]	0.91	0.02	0.61	0.07	0.46	0.05
UG-RNN-CR+LogP[15]	0.91	0.02	0.63	0.05	0.47	0.03
GSE[13]	-	-	-	-	0.47	-
2D Kernel (param d=2)[2]	0.91	-	0.61	-	0.44	-
3D Delaunay[2]	0.88	-	0.72	-	0.51	-
3D Histogram + Gaussian[2]	0.91	-	0.63	-	0.45	-

Results

Metrics

In order to assess the performance of the NN-PIF predictors and compare them with other methods, we use three standard metrics: the root mean square error (RMSE), the average absolute error (AAE), and the Pearson correlation coefficient (R) defined by

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{i}(t_i - p_i)^2} \tag{7}$$

$$AAE = \frac{1}{n}\sum_{i=1}^{n}|t_i - p_i| \tag{8}$$

$$R = \frac{\sum_{i=1}^{n}(t_i - \bar{t})(p_i - \bar{p})}{\sqrt{\sum_{i=1}^{n}(t_i - \bar{t})^2}\sqrt{\sum_{i=1}^{n}(p_i - \bar{p})^2}} \tag{9}$$

Here p_i is the predicted value and t_i is the target value (experimentally observed) for molecule i. We use R^2 instead of R as our error metric in order to compare our results with other published results. In the tables, for clarity purposes the best results are marked in bold.

Results obtained by 10-fold cross validation on the Small Delaney Dataset are shown in Table 2. NN-PIF performs slightly worse than 2D based methods and 3D histogram + Gaussian but matches the results obtained with 3D Delunay.

Results obtained by 10-fold cross validation on the Huuskonen dataset are shown in Table 3. NN-PIF matches the perfomances of both 2D and 3D based methods and outperforms the results obtained with 3D Delunay.

Results obtained by 10-fold cross validation on the Karthikeyan dataset are shown in Table 4. NN-PIF performs worse than 2D based methods. However it matches the performances of 3D Delunay and outperforms 3D Histogram +

Table 3: Prediction performances and standard deviations using 10-fold cross validation on the Huuskonen Dataset (1026 molecules)

Models	R^2	std R^2	RMSE	std RMSE	AAE	sdt AAE
NNPIF	0.90	0.01	0.65	0.06	0.49	0.04
UG-RNN[15]	**0.91**	0.01	**0.60**	0.06	**0.46**	0.04
UG-RNN-CR[15]	0.80	0.04	0.92	0.07	0.65	0.05
UGR-NN+LogP[15]	**0.91**	0.01	0.61	0.06	0.47	0.04
UG-RNN-CR+LogP[15]	0.89	0.02	0.68	0.06	0.52	0.04
3D Delaunay[2]	0.88	-	-	-	-	-
3D histogram[2] + Gaussian	**0.91**	-	-	-	-	-
RBF Kernel[10]	0.90	-	-	-	-	-

Table 4: Prediction performance in 10 fold cross validation on Karthikeyan Dataset (4173 compounds)

Models	R^2	std R^2	RMSE	std RMSE	AAE	sdt AAE
NNPIF	0.49	0.04	45.92	1.86	35.9	1.34
UGRNN[16]	**0.56**	-	**42.6**	-	33.2	-
2D Kernel[2]	**0.56**	-	42.71	-	**32.58**	-
3D Delaunay[2]	0.50	-	46.62	-	35.01	-
3D histogram + Gaussian[2]	0.27	-	55.01	-	43.38	-
Karthikeyan 2D[14]	0.44	-	49.3	-	38.2	-
Karthikeyan 3D[14]	0.30	-	55.5	-	45.6	-

Gaussian. Moreover, it is important to notice that NN-PIF outperforms both Karthikeyan 2D and Karthikeyan 3D.

Discussion

Our tests show that NN-PIF generally matches, though seldom outpermorms, the perfomances of other 3D based predictors but obtains generally worse results than 2D based predictors (with the only exception of Karthikeyan 2D). This is in agreement with the findings of Azencott et al.[2] and Karthikeyan[14]. The latter does not provide an explanation, simply reporting that calculated 3D descriptors contain less relevant information than 2D descriptors. On the other hand, Azencott et al. observe that the 3D structures of the molecules, which are required to build the predictive models, are not present in the data sets, nor is any information about stereochemistry. The coordinates of the atoms in these structures are predicted and this is likely to introduce errors that affect the performance of molecular property predictors. As far as we know, this is the most likely explanation. Finally, it is important to notice how NN-PIF, which are based on a very simple algorithm that automatically extracts features from the 3D molecular graph, outperform classifiers that use sets of pre-computed descriptors as in Kathikeyan[14]. This is further proof that methods based on a target-driven training process (like UG-RNN[15] for example) show good generalization capabilities with the advantage of reducing drastically the time required by the feature selection step.

Acknowledgements

AL is funded through a GREP Ph.D. Scholarship by the Irish Research Council for Science, Engineering and Technology. GP's research is partly funded through Science Foundation Ireland RFP grant 10/RFP/GEN2749.

References

[1] Marvin beans.

[2] C-A Azencott, A Ksikes, SJ Swamidass, JH Chen, L Ralaivola, and P Baldi. One- to four-dimensional kernels for virtual screening and the prediction of physical, chemical, and biological properties. *J. Chem. Inf. Comput. Sci*, 47:965–974, 2007.

[3] P Baldi, S Brunak, P Frasconi, G Soda, and G Pollastri. Exploiting the past and the future in protein secondary structure prediction. *Bioinformatics*, 15(11):937–946, 1999.

[4] P Baldi and G Pollastri. The principled design of large-scale recursive neural network architectures-dag-rnns and the protein structure prediction problem. *Journal of Machine Learning Research*, 4:575–602, 2003.

[5] C Bergström, U Norinder, K Luthman, and P Artursson. Molecular descriptors influencing melting point and their role in classification of solid drugs. *Journal of Chemical Information and Modeling*, 43:1177–1185, 2003.

[6] D Cherqaoui and D Villemin. Use of neural network to determine the boiling point of alkanes. *J. Chem. Soc.*, 90:97–102, 1994.

[7] Syracuse Research Corporation. Physical/chemical property database(physoprop). SRC Environmental Science Center: Syracuse, NY, 1994.

[8] JS Delaney. Esol: Estimating aqueous solubility directly from molecular structure. *J. Chem. Inf. Comput. Sci*, 44:1000–1005, 2003.

[9] H Fróhlich, J Wegner, and A Zell. Towards optimal descriptor subset selection with support vector machines in classification and regression. *Journal of Chemical Information and Modeling*, 45(3), 2005.

[10] H Fröhlich, J K Wegner, and A Zell. Towards optimal descriptor subset selection with support vector machines in classification and regression. *QSAR & Combinatorial Science*, 23(5):311–318, 2004.

[11] L. Hanses and L. Salamon. Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:993–1001, 1990.

[12] J Huuskonen. Estimation of aqueous solubility for a diverse set of organic compounds based on molecular topology. *J. Chem. Inf. Comput. Sci*, 40:773–777, 2000.

[13] N Jain and SH Yalkowsky. *J. Pharm. Sci*, 90:234–252, 2001.

[14] M Karthikeyan. General melting point prediction based on a diverse compound data set and artificial neural networks. *Journal of Chemical Information and Modeling*, 45:581–590, 2005.

[15] A Lusci, G Pollastri, and P Baldi. Deep architectures and deep learning in chemoinformatics the prediction of aqueous solubility for drug like molecules. *Journal of Chemical Information and Modeling*, 2013.

[16] A Lusci, I Walsh, and G Pollastri. Adaptive virtual screening of drug-like molecules by recursive neural networks for undirected. graphs. IEEE: Proc. 6th International Conference on Bioinformatics and Biomedical Engineering, Shanghai, China, May 2012.

[17] AJM Martin, C Mirabello, and G Pollastri. Neural network pairwise interaction fields for protein model quality assessment and ab initio protein folding. *Current Protein & Peptide Science*, 12(6):549–562, 2011.

[18] C Mirabello and G Pollastri. Porter, paleale 4.0: high-accuracy prediction of protein secondary structure and relative solvent accessibility. *Bioinformatics*, 2013.

[19] C Mooney and G Pollastri. Beyond the twilight zone: Automated prediction of structural properties of proteins by recursive neural networks and remote homology information. *Proteins*, 77(1):181–90, 2009.

[20] C Mooney, A Vullo, and G Pollastri. Protein structural motif prediction in multidimensional ø-ψ space leads to improved secondary structure prediction. *Journal of Computational Biology*, 13(8):1489–1502, 2006.

[21] C Mooney, Y H Wang, and G Pollastri. Sclpred: protein subcellular localization prediction by n-to-1 neural networks. *Bioinformatics*, 27(20):2812–9, 2011.

[22] G Pollastri and P Baldi. Prediction of contact maps by recurrent neural network architectures and hidden context propagation from all four cardinal corners. *Bioinformatics*, 18, Suppl.1:S62–S70, 2002.

[23] G Pollastri and A McLysaght. Porter: a new, accurate server for protein secondary structure prediction. *Bioinformatics*, 21(8):1719–1720, 2005.

[24] N Qian. On the momentum term in gradient descent learning algorithms. *Neural Networks*, 12(1):145–151, 1999.

[25] A. Starita, A. Micheli, and A. Sperduti. Analysis of the internal representations developed by neural networks for structures applied to quantitative structure-activity relationship studies of benzodiazepines. *Journal of Chemical Information and Modeling*, 41:202–218, 2000.

[26] V Volpato, A Adelfio, and G Pollastri. Accurate prediction of protein enzymatic class by n-to-1 neural networks. *BMC Bioinformatics*, 14(S1):S11, 2013.

[27] A Vullo, I Walsh, and G Pollastri. A two-stage approach for improved prediction of residue contact maps. *BMC bioinformatics*, 7(1):180, 2006.

[28] I Walsh, D Baù, AJM Martin, C Mooney, A Vullo, and G Pollastri. Ab initio and template-based prediction of multi-class distance maps by two-dimensional recursive neural networks. *BMC structural biology*, 9(1):5, 2009.

[29] SH Yalkowsky and RM Dannelfelser. The arizona database of aqueous solubility. College of Pharmacy, University of Arizona: Tucson, AZ, 1990.