Pairwise Input Neural Network for Target-Ligand Interaction Prediction

Caihua Wang, Juan Liu*, Fei Luo and Yafang Tan
School of Computer
Wuhan University
Wuhan, PR China 430072
Email: liujuan@whu.edu.cn

Zixin Deng and Qian-Nan Hu*
Key Laboratory of Combinatorial Biosynthesis
and Drug Discovery (Ministry of Education)
and Department of Pharmaceutical Sciences,
Wuhan University, Wuhan, PR China 430072
Email: qnhu@whu.edu.cn

***Abstract*—Prediction the interactions between proteins (targets) and small molecules (ligands) is a critical task for the drug discovery in silico. In this work, we consider the target binding site instead of the whole target and propose a pairwise input neural network (PINN) for constructing the site-ligand interaction prediction model. Different with the ordinary artificial neural network (ANN) with one vector as input, the proposed PINN can accept a pair of vectors as the input, corresponding to a binding site and a ligand respectively. The 5-CV evaluation results show that PINN outperforms other representative target-ligand interaction prediction methods.**

I. Introduction

Predicting interactions between small molecules and proteins plays a critical role in drug discovery *in silico*. Through various high-throughput experimental projects for analyzing the genome, transcriptome and proteome, we are beginning to understand the genomic spaces. At the same time, the high-throughput screening of large-scale chemical compound libraries with various biological assays is enabling us to explore the chemical space [1], [2]. However, our knowledge about the relationship between the chemical and genomic spaces is very limited [3], [4]. Since experimental ways to determine target-ligand interaction are costly and time-consuming, there is a strong incentive to develop computational methods capable of predicting potential target-ligand interactions.

In recent years, researchers have began to made great efforts to construct the prediction models. Some researchers represented the targets based on the full length protein sequence information, encoded the ligands based on PubChem fingerprints [5] and built the pairwise support vector machine (pSVM) to predict the target-ligand interactions [6], [7]. Following the similar framework, Jacob *et al.* [8] also constructed a prediction model, except that they adopted EC (Enzyme Commission) numbers rather then the protein sequences themselves to describe the targets. Laarhoven *et al.* [9] firstly extracted the target profiles from the target-ligand interaction network and integrated them with protein sequences information to generate a target kernel, next extracted the the ligand profiles from the target-ligand interaction network and integrated them with the molecular graphs (SIMCOMP [10]) to generate a ligand kernels, after that a pairwise kernel was applied to combine the target and ligand kernel, Finally, the the kernel ridge regression (RLS) was employed to construct a prediction model. Yamanishi *et al.* integrated the chemical space and genomic space into a unified space ("pharmacological space") and developed a supervised learning algorithm to infer unknown drug-target interactions [4]. Bleakley and Yamanishi proposed a bipartite local model (BLM) for the interaction prediction [11] and achieved good perforamnces. After that, Mei *et al.* improved the BLM with neighbor based interaction-profile inferring (BLM_NII) and achieved the state-of-the-art performance in Yamanishi's "golden standard" data set [12]. Yamanishi *et al.* attempted to find the potential inherent factors governing the target-ligand interactions by extracting features between chemical substructures and protein domains (CS-PD) and applied a simple threshold to predict target-ligand interaction [10].

Most of the published papers consider the protein target as a whole, however, not all parts of the proteins involve in the target-ligand interaction. In fact, ligand binding proteins usually contain one or more binding sites for substrates or for activators/inhibitors. Taking the non-binding sites information into considered may introduce noises into the prediction models. Therefore, in this work we only consider the binding sites of the targets and extract the site-ligand information to construct the interaction prediction model. Traditional supervised classification algorithms, except for pairwise kernel methods such as pSVM, only accept a vector as input, but we desire for a classification algorithm accepting two input vectors in the site-ligand interaction setting. The pairwise kernel in pSVM maps the target and ligand features into a high-dimensional space, but we know little about the feature interactions in high-dimensional space, which leads to bad chemical/biology interpretation of the model. In this work, the proposed pairwise input neural network (PINN) can overcome the above disadvantages. The PINN can not only accept two vectors as input but also preforms linear feature learning in the lower layer , which is interpretable in chemical. Moreover, inspired by the good performance of deep learning in other application areas, the training of PINN includes pre-training and fine tuning.

II. Pairwise input neural network

A. Problem description

Assume the number of all the binding sites is p, and every binding site is represented with an m-dimension vector, we denote the binding sites with a matrix $\boldsymbol{S} = (\boldsymbol{s_1}, \boldsymbol{s_2}, \boldsymbol{s_3}, ..., \boldsymbol{s_p})^T$, where $\boldsymbol{s_i} = (s_{1i}, s_{2i}, s_{3i}, ..., s_{mi})^T$ describes the $\boldsymbol{i}$-th binding site. Assume the number of all the ligands is q, and every

978-1-4799-5669-2/14/$31.00 ©2014 IEEE

ligand is represented with an n-dimension vector, we denote the ligands as a matrix $\boldsymbol{L} = (\boldsymbol{l_1}, \boldsymbol{l_2}, \boldsymbol{l_3}, ..., \boldsymbol{l_q})^T$, where $\boldsymbol{l_j} = (l_{1j}, l_{2j}, l_{3j}, ..., l_{nj})^T$ encodes the $\boldsymbol{j}$-th ligand. Without loss of generality, we assume the columns of S and L have been centered and scaled. The interaction pairs are denoted as $\boldsymbol{D} = ((i_1, j_1), (i_1, j_2), \cdots, (i_N, j_N))$, where N is the sample size of the data set, $i_k \in \{1, 2, \cdots, p\}$, $j_k \in \{1, 2, \cdots, q\}$, and (i_k, j_k) represents the $\boldsymbol{k}$-th site-ligand pair. Correspondingly, we use $\boldsymbol{y} = (y_1, y_2, \cdots, y_N)$ to represent the binding label vector for D, where $y_k \in \{0, 1\}$ is the binding label of site-ligand pair (i_k, j_k).

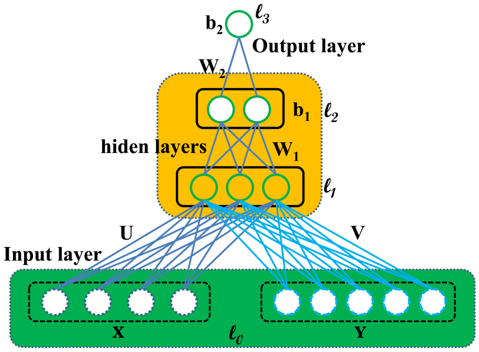

Fig. 1. Pairwise input neural network (PINN). PINN could accept a pair of vectors as input and output the predicted interaction confidence.

B. Definition of PINN

Traditional supervised classification algorithms, only accept a vector as input, but we desire for a classification algorithm accepting two input vectors in the site-ligand interaction setting. In order to construct the prediction model, we propose the PINN as Figure 1. The PINN could contains more than four layers, but we only discuss four layers, l_0, l_1, l_2, l_3 (ordered form the input layer to the output layer), in this work. The input layer (l_0) in PINN contains two separate parts, X and Y. The weights between X and l_1 are $\boldsymbol{U}$, and the weights between Y and l_1 are $\boldsymbol{V}$. The weights between layer l_i and layer l_{i+1} are denoted as $\boldsymbol{W}_i$ $(i \geq 1)$. The biases of layer l_i $(i > 1)$ are denoted as $\boldsymbol{b}_{i-1}$. The PINN in Figure 1 could be represented as following:

$$H_f(\boldsymbol{s}, \boldsymbol{l}; \boldsymbol{\theta}) = f(\boldsymbol{W_2} f(\boldsymbol{W_1}((\boldsymbol{Us}) \circ (\boldsymbol{Vl})) + \boldsymbol{b_1}) + \boldsymbol{b_2}) \quad (1)$$

where $\boldsymbol{\theta} = ((\boldsymbol{U}, \boldsymbol{V}), (\boldsymbol{W}_1, \boldsymbol{b}_1), (\boldsymbol{W}_2, \boldsymbol{b}_2))$, f and $\circ$ represent the transfer function and element wise multiply respectively. $\boldsymbol{s}, \boldsymbol{l}$ represent the target binding site and ligand respectively. If $H_f(\boldsymbol{\theta}; \boldsymbol{s}, \boldsymbol{l})$ larger than some threshold (0 or 0.5, depending on the choice of the transfer function), we predict positive interaction, otherwise predict negative. To fit the parameters, we adopt the square loss function and L_2 norm regularization, which could help to avoid overfitting. As a result, the overall cost function, $J(\boldsymbol{\theta})$, is defined as following:

$$J(\boldsymbol{\theta}) = \frac{1}{2N} \sum_{(i_k, j_k) \in D} (H_f(\boldsymbol{s_{i_k}}, \boldsymbol{l_{j_k}}; \boldsymbol{\theta}) - y_k)^2 + \frac{\eta}{2} \|\boldsymbol{\theta}\|^2 \quad (2)$$

where $\boldsymbol{s_{i_k}}$ and $\boldsymbol{l_{j_k}}$ denote the site and ligand respectively in the $\boldsymbol{k}$-th site-ligand pair (i_k, j_k).

C. Training of PINN

The cost function of PINN is non-convex, therefore the initial parameters are of great importance to the ultimate classification performance. Hinton *et al.* proposed pre-training and fine tuning procedure to train deep neural network to avoid the local optimum [13]. The pre-training procedure can provide good initial values for parameters and fine tuning procedure can improve the performance of neural network. PINN, processing more than three layers, is a kind deep network and we follow the pre-training and fine tuning procedure. The basic algorithm is shown in Algorithm 1.

Algorithm 1 Pairwise Input Neural Network (PINN)

```
Pre-training:
    1) Pre-training for the first layer
        a) Calculate a matrix:
            $\boldsymbol{A} = \sum_{(i_k, j_k) \in \boldsymbol{D}} d(i_k, j_k) \boldsymbol{s_{i_k}} \boldsymbol{l_{j_k}^T}$
        b) Apply SVD to decompose $\boldsymbol{A}$ to initiate $\boldsymbol{U}, \boldsymbol{V}$:
            $\boldsymbol{A} = \widehat{U} \wedge \widehat{V}^T,\ \boldsymbol{U} = \widehat{U} \wedge^{1/2},\ \boldsymbol{V} = \widehat{V} \wedge^{1/2}$
    2) Pre-training the hidden layers with RBM/AutoEncode.
Fine tune:
    Apply the back propagation algorithm to fine tune the whole network.
```

The hidden layer parameters of PINN (for example, $\boldsymbol{W}_1$ in Figure 1) could be initiated by RBM [13] or AutoEncode without any modification. However, there is no existing approach to initiate the pairwise input layer parameters (for example, $\boldsymbol{U}, \boldsymbol{V}$ in Figure 1) in neural network setting. Therefore, we apply a simple method, canonical correlation analysis (CCA), to initiate $\boldsymbol{U}, \boldsymbol{V}$. Based on CCA, we have

$$\begin{aligned} corr(u, v) &= \frac{\sum_{(i_k, j_k) \in \boldsymbol{D}} d(i_k, j_k)(\boldsymbol{s_{i_k}^T u})(\boldsymbol{l_{j_k}^T v})}{\sqrt{\sum_{i_k} (\boldsymbol{u^T s_{i_k}})^2} \sqrt{\sum_{j_k} (\boldsymbol{v^T l_{j_k}})^2}} \quad (3) \\ &= \frac{\boldsymbol{u}^T (\sum_{(i_k, j_k) \in \boldsymbol{D}} d(i_k, j_k) \boldsymbol{s_{i_k}} \boldsymbol{l_{j_k}^T}) \boldsymbol{v}}{\sqrt{\boldsymbol{u}^T (S^T S) \boldsymbol{u}} \sqrt{\boldsymbol{v}(L^T L) \boldsymbol{v}}} \end{aligned}$$

where:

$$d(i_k, j_k) = \begin{cases} 1 & \text{if site } i_k \text{ interacts with ligand } j_k \\ -\lambda & \text{otherwises} \end{cases}$$

$d(i_k, j_k)$ represents the binding confidence. If site i_k could interact with ligand j_k (observed in the experiment), we set $d(i_k, j_k) = 1$. If we do not know whether site i_k could interact with ligand j_k (the way we generate the negative pairs), we set $d(i_k, j_k) = -\lambda$, $0 \leq \lambda \leq 1$. $\boldsymbol{u}, \boldsymbol{v}$ are parameters in the map matrices $\boldsymbol{U}, \boldsymbol{V}$.

We denote, $\boldsymbol{A} = \sum_{(i_k, j_k) \in \boldsymbol{D}} d(i_k, j_k) \boldsymbol{s_{i_k}} \boldsymbol{l_{j_k}^T}$. Obviously, A is a constant matrix. It has been shown that in high-dimensional problems, treating the covariance matrix as diagonal can yield good results [14]. For this reason, rather than using Eq. (3) as our canonical correlation analysis (CCA) criterion, we substitute in the identity matrix I for $S^T S$ and $Y^T Y$; this gives what could be called “diagonal penalized CCA”. Eq. (3) could be rewrite as,

$$corr(u, v) = \max_{\boldsymbol{u},\boldsymbol{v}} \boldsymbol{u^T A v}, \quad s.t.\ \|\boldsymbol{u}\|_2^2 = 1,\ \|\boldsymbol{v}\|_2^2 = 1 \qquad (4)$$

The optimal solution of Eq. (3) is a rank-1 approximation of matrix $\boldsymbol{A}$, because,

$$\begin{aligned} \|A - duv^T\|_{Fro}^2 &= Tr\left((A - duv^T)^T(A - duv^T)\right) \qquad (5) \\ &= d^2\|u\|_2^2\|v\|_2^2 - 2du^TAv + Tr(A^TA) \end{aligned}$$

If we let $\|u\|_2^2 = 1, \|v\|_2^2 = 1$, the problem in Eq. (4) is equivalent with the following problem and $corr(u, v) = d$.

$$\min \|A - duv^T\|_{Fro}^2, \quad s.t.\ \|\boldsymbol{u}\|_2^2 = 1,\ \|\boldsymbol{v}\|_2^2 = 1 \qquad (6)$$

Based on the relationship between singular value decomposition (SVD) and rank-r matrix approximation,

$$\sum_{k=1}^{r} d_k u_k v_k^T = \arg \min_{\widehat{A} \in M(r)} \|A - \widehat{A}\|_{Fro}^2 \qquad (7)$$

where u_k, v_k is the k-th left and right singular vector of A and d_k is the k-th singular, $M(r)$ is a matrix set with rank r. Therefore, we apply SVD to decompose A to initiate U,V.

$$A = \widehat{U} \wedge \widehat{V}^T,\ U = \widehat{U}\wedge^{1/2},\ V = \widehat{V}\wedge^{1/2} \qquad (8)$$

After pre-training, we apply back propagation algorithm to fine tune PINN. The back propagation algorithm can be divided into two phases: propagation and weight update. In the propagation phase, the input samples go through the neural network in order to generate the output activations and then the training error backward propagate through the neural network to calculate the deltas of all output and hidden neurons. In the weight update phase, multiply the output deltas and input activations to get the gradient of the weight and subtract a ratio (learning rate) of the gradient from the weight.

III. Prediction of Site-Ligand Interaction

A. Data set

We extracted all the target-ligand interactions of human being available from the sc-PDB database [15]. As a result, we got 836 targets (782 single-site targets and 54 multi-site targets) and 2710 different ligands. Totally, there are 6830 target-ligand pairs included in our data set (Table I).

TABLE I. Statistics of data set.

	Target	Ligand	Pairs
One-site	782	1988	5122
Multi-site	54	722	1708
Total	836	2710	6830

B. Representations of Binding Sites and Ligands

For each target, the amino acid residues possessed at least one atom within 8 angstrom around the ligand is considered as a ligand binding site. In order to represent the binding site, we adopted the dictionary defined by Nagamine and Sakakibara [16], [17], which contains 4200 residue triplets of 199 types. As a result, we can represent each binding site as a 199-dimensional vector, and an element of the vector represents the frequency of the corresponding type of triplet occurring in the binding site.

In order to represent the ligands, we first integrated information from PubChem (containing 881 predefined chemical substructures) [5] and Checkmol document [18] into a universal fingerprint set. Then the single atoms and bonds are removed from the set for they are not in the same structural level with site fragments. Some substructures, such as benzene, are too common to serve as discriminately features are also removed from the set. As a result, we got a fingerprint set with 413 chemical substructures. For each ligand, we used the OpenBabel tool [19] to search the fingerprints, and the ligand can then be represented as a 413-dimensional binary vector whose elements represent the presence or absence of each substructure by 1 or 0, respectively.

Once the binding sites and the ligands are represented with 199- and 413- dimensional vectors, we can then build PINN models to predict the site-ligand interactions described in above section.

C. Results and discussion

1) Performance evaluation: In order to evaluate the performances of PINN, we used compare PINN with other two representative approaches by 5-fold cross-validation: BLM-NII [12] and CS-PD [10]. The performances are evaluated by multiple criteria, such as accuracy (ACC), precision, recall and area under receiver operating characteristic curve (AUC). The test results are shown in Table II, illustrating that PINN outperforms CS-PD and BLM-NII.

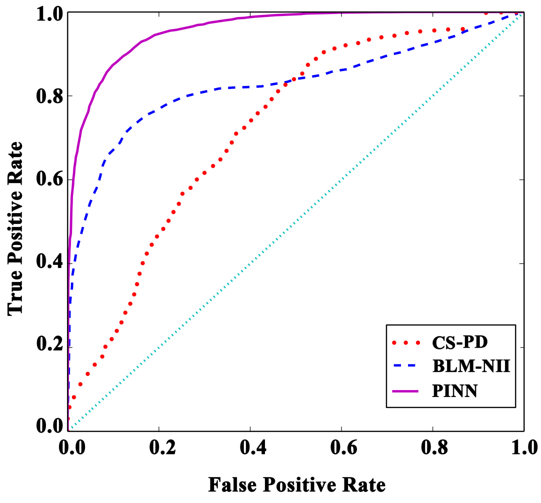

Fig. 2. ROC curves of CS-PD, BLM-NII and PINN. The AUC of CS-PD, BLM-NII and PINN are 0.799, 0.858 and 0.959 respectively. The PINN outperforms the CS-PD and BLM-NII.

TABLE II. Statistics of the prediction performance.

	ACC	Precision	Recall	AUC
CS-PD	0.565	0.552	0.621	0.799
BLM-NII	0.727	0.712	0.812	0.858
PINN	0.887	0.886	0.889	0.959

The ACC and AUC scores of CS-PD are 56.5% and 79.9% respectively, which means the correct prediction rate is only slightly higher than random guess and the comprehensive performance is not good. We guess that the poor performance of CS-PD is due to the lacking of powerful classifier and it only serves as a feature extraction approach. BLM-NII performs well on our data set (AUC 85.8%), but not as well as in its origin data set. The difference of data set could be the main cause of the AUC difference. It is a pity that not all the crystal structures of the targets in Yamanishi‘s data set are determined, and we could not perform our approach in the "Gold Standard". PINN performs excellently in our data set. The AUC scores of PINN is 95.9%, which is much higher than that of CS-PD and BLM-NII. Figure 2 demonstrates detailed ROC curves for CS-PD, BLM-NII and PINN, once again suggesting that PINN outperforms the CS-PD and BLM-NII.

2) Feature network analysis: We only applied a three-layer PINN in this work, which result in interpretable feature interaction network. The underlying mathematics reasons is

$$\begin{aligned} H_f(\boldsymbol{s}, \boldsymbol{l} : \boldsymbol{\theta}) &= f(\boldsymbol{w}((\boldsymbol{U}s^T) \circ (\boldsymbol{V}l^T)) + b) \qquad (9) \\ &= f(\sum_m \boldsymbol{w}_m(\boldsymbol{U}_m \boldsymbol{s}^T \boldsymbol{V}_m \boldsymbol{l}^T) + b) \\ &= f(\boldsymbol{s}\left(\sum_m \boldsymbol{w}_m \boldsymbol{U}_m^T \boldsymbol{V}_m\right)\boldsymbol{l}^T + b) \\ &= f(\boldsymbol{s}(\boldsymbol{U}^T diag(\boldsymbol{w})\boldsymbol{V})\boldsymbol{l}^T + b) \end{aligned}$$

where $\boldsymbol{\theta} = (\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{w}, b)$ are parameters, $H_f(\boldsymbol{s}, \boldsymbol{l}; \boldsymbol{\theta})$ is discriminant function and f is sigmoid function. For convenience, we denote $\boldsymbol{M} = \boldsymbol{U}^T diag(\boldsymbol{w})\boldsymbol{V}$, the Equation (9) could be rewriten as following:

$$H_f(\boldsymbol{s}, \boldsymbol{l}) = f(\boldsymbol{s}\boldsymbol{M}\boldsymbol{l}^T + b) \qquad (10)$$

We did not fit the bias in this work, therefore, the prediction function $sign(H_f(\boldsymbol{s}, \boldsymbol{l}; \boldsymbol{\theta}))$ could be calculated as following:

$$Pred_{H_f}(\boldsymbol{s}, \boldsymbol{l}) = \begin{cases} 1, & \boldsymbol{s}\boldsymbol{M}\boldsymbol{l}^T \geq 0 \\ -1, & \boldsymbol{s}\boldsymbol{M}\boldsymbol{l}^T < 0 \end{cases} \qquad (11)$$

Based on the Equation (11), we find that the matrix $\boldsymbol{M}$ is the fragment interaction network/matrix, where the site fragments are regarded as rows, the ligand fragments are regarded as columns, and the values are the fragment interactions intensity fitted by data. The fragment interaction network might be important in both chemogenomics and drug discovery. If we know the ligand-binding site, we could break it into fragments/features based on the target dictionary. Then, we could refer to the fragment interaction network/matrix and find potentially high affinity ligand fragments/features, which could improve the drug design efficiency and save a lot of time and resources.

IV. Conclusion

In this paper, we proposed a pairwise input neural network, PINN, to predict the target-ligand interaction. Different with most of the existed methods, we no longer considered all target information, but just extracted and described the ligand-binding sites to build the prediction model, which enables us to predict more accurate site-ligand interactions. By comparing with other two representative methods, PINN achieved the best performances in terms of multiple criteria, suggesting that the proposed method is promising in drug discovery.

Acknowledgment

This work was supported by the National Science Foundation of China [61272274, 60970063, 31270101]; the program for New Century Excellent Talents in Universities [NCET-10-0644]; the Ministry of Science and Technology of China (973 and 863 Programs) and the National Mega Project on Major Drug Development.

References

[1] M. Kanehisa, S. Goto, M. Hattori, K. F. Aoki-Kinoshita, M. Itoh, and S. Kawashima, "From genomics to chemical genomics: new developments in kegg," *Nucleic Acids Res*, vol. 34 (Database issue), pp. D354–D357, 2006.

[2] B. R. Stockwell, "Chemical genetics: ligand-based discovery of gene function," *Nat Rev Genet*, vol. 1, pp. 116–125, 2000.

[3] D. Rognan, "Chemogenomic approaches to rational drug design," *Br J Pharmacol*, vol. 152, pp. 38–52, 2007.

[4] Y. Yamanishi, M. Araki, A. Gutteridge, W. Honda, and M. Kanehisa, "Prediction of drug-target interaction networks from the integration of chemicaland genomic spaces," *Bioinformatics*, vol. 24, pp. i232–240, 2008.

[5] B. Chen, D. Wild, and R. Guha, "Pubchem as a source of polypharmacology," *J Chem Inf Model*, vol. 49, pp. 2044–2055, 2009.

[6] C. S. Leslie, "Mismatch string kernels for discriminative protein classification," *Bioinformatics*, vol. 20, pp. 467–476, 2004.

[7] J. P. Vert and B. Scholkopf, "Local alignment kernels for biological sequences," *Kernel Methods in Computational Biology, MIT Press*, vol. pp, pp. 131–154, 2004.

[8] L. Jacob, "Protein-ligand interaction prediction: an improved chemogenomics approach," *Bioinformatics*, vol. 24, pp. 2149–2156, 2008.

[9] T. Laarhoven, "Gaussian interaction profile kernels for predicting drug-target interaction," *Bioinformatics*, vol. 27, pp. 3036–3043, 2011.

[10] L. Yamanishi, "Extracting sets of chemical substructuresand protein domains governing drug-target interactions," *J Chem Inf Model*, vol. 51, pp. 1183–1194, 2011.

[11] K. Bleakley and Y. Yamanishi, "Supervised prediction of drug-target interactions using bipartite local models," *Bioinformatics*, vol. 25, pp. 2397–2403, 2009.

[12] J. P. Mei, C. K. Kwoh, P. Yang, X. L. Li, and J. Zheng, "Drug-target interaction prediction by learning from local informationand neighbors," *Bioinformatics*, vol. 29, pp. 238–245, 2013.

[13] I. Sutskever, G. E. Hinton, and G. W. Taylor, "The recurrent temporal restricted boltzmann machine," *NIPS 2008*, pp. 1601–1608, 2009.

[14] N. B. Tibshirani R., Hastie T. and Chu, "Class prediction by nearest shrunken centroids, with applications to dna microarrays." *Statistical Science*, vol. 18, pp. 104–117, 2003.

[15] J. Meslamani, D. Rognan, and E. Kellenberger, "sc-pdb: a database for identifying variationsand multiplicity of 'druggable' binding sites in proteins," *Bioinformatics*, vol. 27, pp. 1324–1326, 2011.

[16] N. Nagamine, "Statistical prediction of protein chemical interactions based on chemical structureand mass spectrometry data," *Bioinformatics*, vol. 23, pp. 2004–2012, 2007.

[17] S. Martin, D. Roe, and J. L. Faulon, "Predicting protein-protein interactions using signature products," *Bioinformatics*, vol. 21, pp. 218–226, 2005.

[18] N. Haider, "Functionality pattern matching as an efficient complementary structure/reaction search tool: an open-source approach," *Molecules*, vol. 15, pp. 5079–1592, 2010.

[19] N. M. OLBoyle, M. Banck, C. A. James, C. Morley, T. Vandermeersch, and G. R. Hutchison, "Open babel: An open chemical toolbox," *J Cheminf*, vol. 3, p. 33, 2011.