DeepDTA: Deep Drug-Target Binding Affinity Prediction

Hakime Öztürk¹, Elif Ozkirimli²*, and Arzucan Özgür¹,*

1 Department of Computer Engineering, Bogazici University, Istanbul, 34342, Turkey
2 Department of Chemical Engineering, Bogazici University, Istanbul, 34342, Turkey

* arzucan.ozgur@boun.edu.tr; elif.ozkirimli@boun.edu.tr

Abstract

The identification of novel drug-target (DT) interactions is a substantial part of the drug discovery process. Most of the computational methods that have been proposed to predict DT interactions have focused on binary classification, where the goal is to determine whether a DT pair interacts or not. However, protein-ligand interactions assume a continuum of binding strength values, also called binding affinity and predicting this value still remains a challenge. The increase in the affinity data available in DT knowledge-bases allow the use of advanced learning techniques such as deep learning architectures in the prediction of binding affinities. In this study, we propose a deep-learning based model that uses only sequence information of both targets and drugs to predict DT interaction binding affinities. The few studies that focus on DT binding affinity prediction either use 3D structure of protein-ligand complexes or 2D features of compounds. One novel approach used in this work is the modeling of protein sequences and compound 1D representations with convolutional neural networks (CNNs). The results show that the proposed deep learning based model that uses the 1D representations of targets and drugs is an effective approach for drug target binding affinity prediction. The model in which a high-level representation of a drug is constructed via CNNs and Smith-Waterman similarity is used for proteins achieved the best Concordance Index (CI) performance, outperforming KronRLS, a state-of-the-art algorithm for DT binding affinity prediction, with statistical significance.

Introduction

The successful identification of drug-target interactions (DTI) is a critical step in drug discovery. As the field of drug discovery expands with the discovery of new drugs, repurposing of existing drugs and identification of novel interacting partners for approved drugs is also gaining interest [40]. Until recently, DTI prediction was approached as a binary classification problem [4,7,14,23,41,50,55].
neglecting an important piece of information about protein-ligand interactions, namely the binding affinity values. Binding affinity provides information on the strength of the interaction between a drug-target (DT) pair and it is usually expressed in measures such as dissociation constant ($K_d$), inhibition constant ($K_i$), or the half maximal inhibitory concentration (IC50). IC50 depends on the concentration of the target and ligand \[9\] and low IC50 values signal strong binding. Similarly, low $K_i/K_d$ values indicate high binding affinity (i.e. good inhibitors have around $K_i$ 1nM or lower). $K_i/K_d$ values are usually represented in terms of $pK_d$ or $pK_i$, the negative logarithm of the binding or inhibition constants.

In binary classification based DTI prediction studies, construction of the data sets constitutes a major problem, since negative (not-binding) information is generally not provided. In most cases, the DT pairs for which binding information is not known are treated as negative (not-binding) samples. The lack of true-negative samples and how the study deals with the generation of synthetic negative samples usually affects the performance of the prediction algorithms. On the other hand, formulating the DT prediction problem as binding affinity prediction, enables the creation of more realistic data sets, where the binding affinity scores are directly used, obviating the need for the generation of synthetic negative samples.

Prediction of protein-ligand interaction binding affinities has been the focus of protein-ligand scoring, which is frequently used after virtual screening and docking campaigns in order to predict the putative strengths of the proposed ligands to the target for identifying the active and inactive compounds \[43\]. Non-parametric machine learning methods such as the Random Forest (RF) algorithm have been used as a successful alternative to parametric scoring functions as of the last decade in order to prevent dependency on the parameters \[3,36,45\]. Later, Gabel et al. showed that RF-score failed in virtual screening and docking tests, speculating that using features such as co-occurrence of atom-pairs oversimplified the description of the protein-ligand complex and led to the loss of information that the raw interaction complex could provide \[20\]. Around the same time this study was published, deep learning started to become a popular architecture powered by the increase in data and high capacity computing machines challenging machine learning methods.

Inspired by the remarkable success rate in image processing \[13,17,46\], and speech recognition \[15,25,30\], deep learning methods are now being exhaustively used in many other research fields, including bioinformatics such as in genomics studies \[35,54\] and quantitative-structure activity relationship (QSAR) studies in drug discovery \[37\]. The major advantage of deep learning architectures is that they enable better representations of the raw data by non-linear transformations in each layer \[34\] and thus, learning hidden patterns from the data.

A few studies employing Deep Neural Networks (DNN) have already been proposed for DTI binary class prediction using different input models for proteins and drugs \[10,26,49\] as well as the ones that employ stacked auto-encoders \[52\] and deep-belief networks \[53\]. Similarly, stacked auto-encoder based models with Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) were applied to represent chemical and genomic structures in real-valued vector
forms. Deep learning approaches have also been applied to protein-ligand interaction scoring in which a common application has been the use of CNNs that learn from the 3D structures of the protein-ligand complexes. However, this approach is limited to known protein-ligand complex structures, with only 25000 ligands reported in the PDB.

Recently, the SimBoost method was proposed to predict binding affinity scores with a gradient boosting machine by using feature engineering to represent drug-target interactions. They utilized similarity-based information of DT pairs as well as features that were extracted from network-based interactions between the pairs. Pahikkala et al., on the other hand, employed Kronecker Regularized Least Squares (KronRLS) algorithm that utilized only similarity-based representations of the drugs and targets using a 2D-based compound similarity method and the Smith-Waterman algorithm, respectively. Both studies used traditional machine learning algorithms and utilized 2D-representations for compounds in order to provide similarity information.

In this study, we propose an approach to predict the binding affinities of protein-ligand interactions with deep learning models using only sequences (1D representations) of proteins and ligands. To this end, the sequences of the proteins and SMILES (Simplified Molecular Input Line Entry System) representations of the compounds are used rather than external features or 3D-structures of the binding complexes that might limit the data set. We employ CNN blocks to learn better representations from the raw protein sequences and SMILES strings and combine these representations to feed into a fully-connected layer block that we termed as DeepDTA. We used the Davis Kinase binding affinity data set to evaluate the performance of our model and compared our results with the KronRLS algorithm.

Our results showed that the model that uses two separate CNN-based blocks to represent proteins and drugs performed as well as the KronRLS algorithm. The model that uses a CNN-block to learn from SMILES and S-W similarity based protein representation, achieved the highest performance with a Concordance Index (CI) of 0.894, significantly outperforming the KronRLS algorithm (0.871) on the task of predicting binding affinities of DT pairs. It also performed significantly better than KronRLS in the task of binding affinity prediction of novel drugs for known proteins.

Materials and Methods

Data set

We evaluated our proposed model on the Kinase data set by as suggested by to be used as benchmark data set for binding affinity prediction evaluation. The Davis data set contains selectivity assays of the kinase protein family and the relevant inhibitors with their respective disassociation constant ($K_d$) values. The data set comprises interactions of 442 proteins and 68 ligands, as reported in Table 1.

The final aim of the model is to predict binding affinity values. While
Table 1. Data set

| Proteins | Compounds | Interactions |
|----------|-----------|--------------|
| Davis ($K_d$) | 442 | 68 | 30056 |

Pahikkala et al. used the $K_d$ values of the Davis data set directly, we used the values transformed into log space $pK_d$ similarly to [28] as explained in Equation 1.

$$pK_d = -\log_{10}(\frac{K_d}{1e9})$$

Figure 1 illustrates the distribution of the binding affinity values in $pK_d$ form. We can clearly observe the peak at $pK_d$ value 5 (10000nM) which constitutes more than half of the data set (20931 out of 30056). These values correspond to the negative pairs that either have very weak binding affinities ($K_d > 10000nM$) or are not observed in the primary screen [42].

The compound SMILES strings were extracted from the Pubchem compound database based on their Pubchem CIDs [5]. Figure 2A illustrates the distribution of the lengths of the SMILES strings of the compounds in the Davis data set. The maximum length of a SMILES is 103, while the average length is equal to 64.

The protein sequences of the Davis data set were extracted from the UniProt protein database based on gene names/RefSeq accession numbers [2]. Figure 2B shows the lengths of the sequences of the proteins in the Davis data set. The maximum length of a protein sequence is 2549 and the average length is 788 characters.

Input Representation

We experimented with two input representation approaches that have been commonly used by deep-learning based studies: one-hot encoding and integer/label encoding. One-hot encoding is a way of representing categorical variables in a binary vector form. For a given set of categories, the entry in a binary vector is set to 1 for the corresponding label and it is set 0 otherwise. We scanned through
Figure 2. Summary of the Davis data set. A) Distribution of the lengths of the SMILES strings B) Distribution of the lengths of the protein sequences

approximately 2M SMILES sequences that we collected from Pubchem and compiled 64 labels (unique letters). For protein sequences, we scanned 550K protein sequences from UniProt and 25 categories (unique letters) were extracted. The example below illustrates the one-hot representation of an example SMILES that belongs to methyl isocyanate, “CN=C=O”. For each character in the SMILES, the corresponding position is set to 1.

```
[*] [C H N 1 O ...n c = +]
[C] [1 0 0 0 0 ...0 0 0 0]
[N] [0 0 1 0 0 ...0 0 0 0]
[=] [0 0 0 0 0 ...0 0 1 0]
[C] [1 0 0 0 0 ...0 0 0 0]
[O] [0 0 0 0 1 ...0 0 0 0]
```

Another popular form input representation is to use integers for the categories (label/integer encoding). Here we simply represent each label with a corresponding integer (e.g. “C”:1, “H”:2, “N”:3 etc.). Label encoding for the example SMILES, “CN=C=O”, is given below.

```
[C N = C = O] = [1 3 63 1 63 5]
```
Similar to the SMILES, protein sequences are encoded in the same fashion using both one-hot and label encodings. Both SMILES and protein sequences have varying lengths. Hence, in order to create an effective representation form, we decided on fixed maximum lengths of 85 for SMILES and 1200 for protein sequences. We chose these maximum lengths based on the distributions illustrated in Figure 2 so that the maximum lengths cover most of the data set. The sequences that are longer than the maximum length are truncated, whereas shorter sequences are 0-padded.

Proposed Model

In this study we treated protein-ligand interaction prediction as a regression problem by aiming to predict the binding affinity scores. As a prediction model, we adopted a popular deep learning architecture, Convolutional Neural Network (CNN). CNN is an architecture that contains one or more convolutional layers often followed by a pooling layer. A pooling layer down-samples the output of the previous layer and provides a way of generalization of the features that are learned by filters. On the top of the convolutional and pooling layers, the model is completed with one or more fully connected layers (FC). The most powerful feature of the CNN models is their ability to capture the local dependencies with the help of filters. Therefore, the number and size of the filters in a CNN directly affects what kind of features the model learns from the input. It is often reported that as the number of filters increases, the model becomes better at recognizing patterns.

We proposed a CNN-based prediction model that comprises two separate CNN blocks, each of which aims to learn representations from SMILES strings and protein sequences. For each CNN block, we used three consecutive 1D-convolutional layers with increasing number of filters. The second and the third convolutional layers had double and triple number of filters the first one had, respectively. The convolutional layers were then followed by max-pooling layer. The final features of the max-pooling layers were concatenated and fed into three fully-connected (FC) layers, that we named as DeepDTA. We used 1024 nodes in the first two FC layers, each followed by a dropout layer of rate 0.1. Dropout is a regularization technique that is used to avoid over-fitting by setting the activation of some of the neurons to 0 [48]. The third layer consisted of 512 nodes and was followed by the output layer. The proposed model that combines two CNN blocks is illustrated in Figure 3.

As activation function, we used Rectified Linear Unit (ReLU) [38], \[ g(x) = \max(0, x) \], since it has been widely used in deep learning studies [34]. A learning model tries to minimize the difference between the expected (real) value and the prediction during training. Since we work on a regression task, we employed mean squared error (MSE) as loss function, in which \( P \) is the prediction vector, whereas \( Y \) corresponds to the vector of actual outputs. \( n \) indicates the number of samples.

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (P_i - Y_i)^2
\]  
(2)
Figure 3. DeepDTA model with two CNN blocks to learn from compound SMILES and protein sequences.

The learning was completed with 100 epochs and mini-batch size of 256 was used to update the weights of the network. Adam was used as the optimization algorithm to train the networks [33] with the default learning rate of 0.001. We also compared two input representation techniques, one-hot and label encoding, therefore we experimented with two ways of feeding data into the prediction system. With one-hot encoding, we directly fed the encoded data into the model, whereas in label-encoding we used Keras’ Embedding layer to represent characters with 128-dimensional dense vectors. The input consisted of (85,128) and (1200, 128) dimensional matrices for the compounds and proteins, respectively.

Results

Baseline

As baseline we chose the model presented by Pahikkala and coworkers where they employed Kronecker Regularized Least Squares (KronRLS) algorithm for binding affinity prediction [42]. KronRLS aims to minimize the following function, where $f$ is the prediction function [42]:

$$J(f) = \sum_{i=1}^{m} (y_i - f(x_i))^2 + \lambda ||f||_k^2$$  \hspace{1cm} (3)

$||f||_k^2$ is the norm of $f$, that is related to the kernel function $k$, and $\lambda > 0$ is a regularization hyper-parameter defined by the user. A minimizer for Equation
can be defined as follows [32]:

\[
f(x) = \sum_{i=1}^{m} a_i k(x, x_i)
\]  (4)

where \(k\) is the kernel function. In order to represent compounds, they utilized a similarity matrix which was computed using SIMCOMP, a tool that utilizes 2D properties of the compounds [27]. As for proteins, the Smith-Waterman algorithm was used to construct a protein similarity matrix [47].

**Evaluation**

To evaluate the performance of a model that outputs continuous values, Concor-dance Index (CI) was used [24]:

\[
CI = \frac{1}{Z} \sum_{\delta_i > \delta_j} h(b_i - b_j)
\]  (5)

where \(b_i\) is the prediction value for the larger affinity \(\delta_i\), \(b_j\) is the prediction value for the smaller affinity \(\delta_j\), \(Z\) is a normalization constant, \(h(m)\) is the step function [42]:

\[
h(x) = \begin{cases} 
1, & \text{if } x > 0 \\
0.5, & \text{if } x = 0 \\
0, & \text{if } x < 0
\end{cases}
\]  (6)

We used paired-t test for the statistical significance tests with 95% confidence interval.

**Experiment Setup**

We evaluated the performance of the proposed model on the Davis data set [16] similarly to [42]. They used nested-cross validation to decide the best parameters for each test set. In order to learn a generalized model, we randomly divided our data set into six equal parts in which one part is selected as the independent test set. The remaining parts of the data set were used to determine the hyper-parameters via five-fold cross validation. Figure 4 illustrates the partitioning of the data set. The same setting was run for [42] for a fair comparison.

![Figure 4. Experiment setup.](image)

We decided on three hyper-parameters for our model, the number of the filters (same for proteins and compounds), the length of the filter size for compounds, and the length of the filter size for proteins. We chose to experiment with different filter lengths for compounds and proteins instead of a common one,
due to the fact that they have different alphabets in terms of characters. The hyper-parameter combination that provided the best average CI score over the five-folds was chosen as the best combination in order to model the test set. We first experimented with hyper-parameters chosen from a wide range and then fine-tuned the model. For example, to determine the number of filters we performed a search over \([16, 32, 64, 128, 512]\). As explained in the Proposed Model subsection, the second convolution layer was set to contain twice the number of filters of the first layer, and the third one was set to contain three times the number of filters of the first layer. 32 filters obtained the best results over the cross-validation experiments. Therefore, in the final model, each CNN block consisted of three 1D convolutions of 32, 64, 96 filters, respectively. For all test results reported in Table 3 we used the same structure summarized in Table 2 except for the lengths of the filters that were used for the compound CNN-block and protein CNN-block.

Table 2. Parameters setting for DTA model

| Parameters                  | Range          |
|-----------------------------|----------------|
| Number of filters           | 32*1; 32*2; 32*3 |
| Filter length (compounds)   | [4,5,6,8]      |
| Filter length (proteins)    | [4,6,8,12]     |
| epoch                       | 100            |
| hidden neurons              | 1024; 1024; 512 |
| batch size                  | 256            |
| dropout                     | 0.1            |
| optimizer                   | Adam           |
| learning rate (lr)          | 0.001          |

In order to provide a more robust performance measure, we evaluated the performance over the independent test set, when the model was trained with the learned parameters in Table 2 on the five training sets that we used in five-fold cross validation (note that the validation sets were not used). The final CI score was reported as the average of these five results. Keras [12] with Tensorflow [1] back-end was used as development framework. Our experiments were run on OpenSuse 13.2 (3.50GHz Intel(R) Xeon(R) and GeForce GTX 1070 (8GB)). The work was accelerated by running on GPU with cuDNN [11].

Performance

In this study, we proposed a deep-learning based model that uses two CNN-blocks to learn representation for drugs and targets using their sequences. As a baseline for comparison, the KronRLS algorithm that uses similarity matrices for proteins and compounds as input was chosen. The Smith-Waterman (S-W) and SIMCOMP algorithms were used to compute the pairwise similarities for the proteins and ligands, respectively. We also illustrated how well the CNN blocks are able to represent proteins and ligands. We first, directly used the S-W and SIMCOMP similarity scores as inputs and fed the combination of these scores to the FC part of our model (DeepDTA), which consists of three
hidden layers and an output layer. We then experimented with two alternative combinations: (i) learning only compound representation with a CNN block and using S-W similarity as protein representation and (ii) learning only protein sequence representation with a CNN block and using SIMCOMP to describe compounds. We also reported the performance of the models that use CNN blocks both with one-hot and categorical representations.

Table 3 reports the average MSE and CI scores over the independent test set of the five models trained with the same parameters (shown in Table 2) using the five different training sets.

Table 3. The average CI and MSE scores over the test set on five different training sets.

|          | Proteins       | Compounds     | CI (std) | MSE   |
|----------|----------------|---------------|----------|-------|
| KronRLS  | Smith-Waterman | SIMCOMP       | 0.871    | 0.379 |
| DeepDTA  | Smith-Waterman | SIMCOMP       | 0.795    | 0.548 |
| DeepDTA  | CNN            | CNN           | 0.871    | 0.297 |
| DeepDTA  | CNN            | CNN           | 0.873    | 0.272 |
| DeepDTA  | CNN            | CNN           | 0.878    | 0.277 |
| DeepDTA  | CNN            | SIMCOMP       | 0.838    | 0.393 |
| DeepDTA  | CNN            | SIMCOMP       | 0.826    | 0.445 |
| DeepDTA  | Smith-Waterman | CNN           | 0.888    | 0.260 |
| DeepDTA  | Smith-Waterman | CNN           | 0.894    | 0.308 |

Using only the fully-connected part of the neural networks (DeepDTA) with S-W and SIMCOMP similarity scores to describe proteins and drugs was outperformed by the baseline, KronRLS algorithm. The combined CNN model that we proposed, on the other hand, performs as well as the baseline with both one-hot and label encoded inputs. The model where only compound representation was built by a CNN block, however, achieved the best CI score with a statistical significance over the baseline with both one-hot and label encoding (p-value=0.0004 and p-value=0.0014, respectively). The MSE values of these models were also significantly less than the MSE of the baseline model.

In the model where only protein representations were built with a CNN block, with both one-hot encoding and label-encoding, the model performed poorly. This might be due to two reasons: i) The CNN model could not effectively learn from amino-acid sequences, and ii) SIMCOMP can not represent compounds as successfully as the SMILES based CNN representation.

One-hot encoding is usually used when there is no ordered relationship between the variables, since label encoding brings in ordinal relationships into the data even if they don’t exist. For the model in which compounds were represented via CNN-based learning and proteins were represented with S-W similarity scores, the difference between the performances of one-hot encoding and label encoding was considered as statistically significant with p-value of 0.012. Despite performing better with one-hot encoding on CI score (ranking) based evaluation metric, we observed that the model produced the smallest MSE value with label-encoded SMILES inputs.

We also used the one-hot encoded CNN and S-W based DeepDTA model to evaluate the performance of a harder DTI prediction problem, which was to
predict new drugs for known proteins. The model (optimized using Stochastic Gradient Descent \cite{6}, lr=0.01) produced an average CI score of 0.701, while the KronRLS algorithm with SIMCOMP and S-W had an average CI score of 0.65, thus outperforming the baseline with statistical significance (p-value=0.035).

For the CNN-based protein sequence and SIMCOMP-similarity based compound representation model, however, we see that the label-encoding model performed better than the one-hot encoding model with a statistical significance (p-value=0.016). The results were indeed complementary to our existing knowledge of amino-acid substitution matrices \cite{29} indicating the order of the amino-acids is important in protein sequences. Therefore, we also tested a combined CNN based model in which amino-acid sequences were represented with label encoding and SMILES were represented as one-hot encoding. The results indicated that CI score (0.878) improved upon both of the homogeneous models, though not significantly.

Table 4. The average CI scores over the test set on five different training sets with one-hot encoding.

| Proteins          | Compounds      | CI (std)     | MSE |
|-------------------|----------------|--------------|-----|
| DeepDTA (SMIlen=103) | Smith-Waterman | CNN          | 0.892 (0.004) | 0.330 |
| DeepDTA (SMIlen=64)  | Smith-Waterman | CNN          | 0.893 (0.003) | 0.274 |
| DeepDTA (SMIlen=85)  | Smith-Waterman | CNN          | 0.894 (0.003) | 0.308 |

As we obtained the best performance with one-hot encoded CNN and S-W combined DTA model, we decided to observe whether the maximum length of the SMILES string affected the performance of the prediction. Table 4 reports the performances of the CNN and S-W combined models when the maximum length of the SMILES was chosen as the length of the longest SMILES (103), the average of SMILES in Davis (64) and our choice (85). We observed that there is not a significant difference between the CI scores.

**Discussion**

In this study, we proposed a deep-learning based approach to predict drug-target binding affinity using only sequences of proteins and drugs. We used Convolutional Neural Networks to learn representations from the raw sequence data of proteins and drugs. We compared the performance of the proposed model with a recent study that employed the KronRLS regression algorithm \cite{42} as our baseline. The model with two CNN-blocks performed as well as the baseline, whereas the model that uses CNN to learn compound representations from SMILES and S-W to compute protein similarity from amino-acid sequences achieved better performance than the KronRLS based algorithm with statistical significance.

After showing that SMILES based compound representation coupled with S-W protein similarity had the highest score in the prediction of drug - target interactions in which the drugs and the targets were present in the training data set, we tested the effectiveness of our methodology on a data set in which the proteins were previously encountered but the drugs were novel. Our model
performed significantly better than the baseline KronRLS algorithm in this prediction task, which requires better representation of the drugs since it aims to predict affinities for novel compounds. This successful performance of the model on the task of predicting affinities for novel drugs supports the effectiveness of the CNN architecture in describing compounds using SMILES strings.

We investigated the effect of the use of different input representation techniques for SMILES and amino-acid sequences on the performance of the proposed models. For SMILES strings, one-hot encoding based SMILES model produced the highest CI score, whereas label encoding based SMILES model produced the lowest mean square error (MSE) value. On the other hand, amino-acid sequences were better represented with label-encoding, which considers the ordinal information of the integers, rather than one-hot encoding. This might be an indication that amino-acids indeed require a structure that can handle their ordered relationships, which the CNN architecture failed to capture successfully. Long-Short Term Memory (LSTM), which is a special type of Recurrent Neural Networks (RNN), could be a more suitable approach to learn from protein sequences, since the architecture has memory blocks that allow effective learning from a long sequence.

The major contribution of this study is the presentation of a novel deep learning-based model for drug-target affinity prediction that uses only character representations of proteins and drugs. SMILES representation for compounds was shown to be effective in predicting affinities for novel compounds. As future work, we focus on building an effective representation for protein sequences. A large percentage of proteins remains untargeted either due to bias in the drug discovery field for a select group of proteins or due to their undruggability and this untapped pool of proteins has gained interest with protein deorphanizing efforts [18, 19, 39]. The methodology can be extended to predict the affinity of known compounds to novel protein targets with no previously identified ligands as well as to the prediction of the affinity of novel drug-target pairs.

Acknowledgments

TUBITAK-BIDEB 2211-E Scholarship Program (to HO) and BAGEP Award of the Science Academy (to AO) are gratefully acknowledged. We thank Ethem Alpaydın, Attila Gürsoy and Pınar Yolum for the helpful discussions.

Funding

This work is funded by Bogazici University Research Fund (BAP) Grant Number 12304.

References

1. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale
machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

2. R. Apweiler, A. Bairoch, C. H. Wu, W. C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, et al. Uniprot: the universal protein knowledgebase. *Nucleic acids research*, 32(suppl_1):D115–D119, 2004.

3. P. J. Ballester and J. B. Mitchell. A machine learning approach to predicting protein–ligand binding affinity with applications to molecular docking. *Bioinformatics*, 26(9):1169–1175, 2010.

4. K. Bleakley and Y. Yamanishi. Supervised prediction of drug–target interactions using bipartite local models. *Bioinformatics*, 25(18):2397–2403, 2009.

5. E. E. Bolton, Y. Wang, P. A. Thiessen, and S. H. Bryant. Pubchem: integrated platform of small molecules and biological activities. *Annual reports in computational chemistry*, 4:217–241, 2008.

6. O. Bousquet and L. Bottou. The tradeoffs of large scale learning. In *Advances in neural information processing systems*, pages 161–168, 2008.

7. D.-S. Cao, S. Liu, Q.-S. Xu, H.-M. Lu, J.-H. Huang, Q.-N. Hu, and Y.-Z. Liang. Large-scale prediction of drug–target interactions using protein sequences and drug topological structures. *Analytica chimica acta*, 752:1–10, 2012.

8. D.-S. Cao, L.-X. Zhang, G.-S. Tan, Z. Xiang, W.-B. Zeng, Q.-S. Xu, and A. F. Chen. Computational prediction of drug–target interactions using chemical, biological, and network features. *Molecular Informatics*, 33(10):669–681, 2014.

9. R. Z. Cer, U. Mudunuri, R. Stephens, and F. J. Lebeda. Ic 50-to-k i: a web-based tool for converting ic 50 to k i values for inhibitors of enzyme activity and ligand binding. *Nucleic acids research*, 37(suppl_2):W441–W445, 2009.

10. K. C. Chan, Z.-H. You, et al. Large-scale prediction of drug-target interactions from deep representations. In *Neural Networks (IJCNN), 2016 International Joint Conference on*, pages 1236–1243. IEEE, 2016.

11. S. Chetlur, C. Woolley, P. Vandermersch, J. Cohen, J. Tran, B. Catanzaro, and E. Shelhamer. cudnn: Efficient primitives for deep learning. *arXiv preprint arXiv:1410.0759*, 2014.

12. F. Chollet et al. Keras, 2015.

13. D. Ciregan, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3642–3649. IEEE, 2012.
14. M. C. Cobanoglu, C. Liu, F. Hu, Z. N. Oltvai, and I. Bahar. Predicting drug–target interactions using probabilistic matrix factorization. *Journal of chemical information and modeling*, 53(12):3399–3409, 2013.

15. G. E. Dahl, D. Yu, L. Deng, and A. Acero. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(1):30–42, 2012.

16. M. I. Davis, J. P. Hunt, S. Herrgard, P. Ciceri, L. M. Wodicka, G. Pallares, M. Hocker, D. K. Treiber, and P. P. Zarrinkar. Comprehensive analysis of kinase inhibitor selectivity. *Nature biotechnology*, 29(11):1046–1051, 2011.

17. J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, pages 647–655, 2014.

18. A. M. Edwards, R. Isserlin, G. D. Bader, S. V. Frye, T. M. Willson, and H. Y. Frank. Too many roads not taken. *Nature*, 470(7333):163, 2011.

19. O. Fedorov, S. Müller, and S. Knapp. The (un) targeted cancer kinome. *Nature chemical biology*, 6(3):166, 2010.

20. J. Gabel, J. Desaphy, and D. Rognan. Beware of machine learning-based scoring functions on the danger of developing black boxes. *Journal of chemical information and modeling*, 54(10):2807–2815, 2014.

21. J. Gomes, B. Ramsundar, E. N. Feinberg, and V. S. Pande. Atomic convolutional networks for predicting protein-ligand binding affinity. *arXiv preprint arXiv:1703.10603*, 2017.

22. R. Gómez-Bombarelli, D. Duvenaud, J. M. Hernández-Lobato, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *arXiv preprint arXiv:1610.02415*, 2016.

23. M. Gönen. Predicting drug–target interactions from chemical and genomic kernels using bayesian matrix factorization. *Bioinformatics*, 28(18):2304–2310, 2012.

24. M. Gönen and G. Heller. Concordance probability and discriminatory power in proportional hazards regression. *Biometrika*, 92(4):965–970, 2005.

25. A. Graves, A.-r. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE, 2013.

26. M. Hamanaka, K. Taneishi, H. Iwata, J. Ye, J. Pei, J. Hou, and Y. Okuno. Cgbvs-dnn: Prediction of compound-protein interactions based on deep learning. *Molecular Informatics*, 2016.
27. M. Hattori, N. Tanaka, M. Kanehisa, and S. Goto. Simcomp/subcomp: chemical structure search servers for network analyses. *Nucleic acids research*, 38(suppl_2):W652–W656, 2010.

28. T. He, M. Heidemeyer, F. Ban, A. Cherkasov, and M. Ester. Simboost: a read-across approach for predicting drug–target binding affinities using gradient boosting machines. *Journal of cheminformatics*, 9(1):24, 2017.

29. S. Henikoff and J. G. Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22):10915–10919, 1992.

30. G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

31. S. Jastrzkeski, D. Lesniak, and W. M. Czarnecki. Learning to smile (s). *arXiv preprint arXiv:1602.06289*, 2016.

32. G. Kimeldorf and G. Wahba. Some results on tchebycheffian spline functions. *Journal of mathematical analysis and applications*, 33(1):82–95, 1971.

33. D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

34. Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

35. M. K. Leung, H. Y. Xiong, L. J. Lee, and B. J. Frey. Deep learning of the tissue-regulated splicing code. *Bioinformatics*, 30(12):i121–i129, 2014.

36. H. Li, K.-S. Leung, M.-H. Wong, and P. J. Ballester. Low-quality structural and interaction data improves binding affinity prediction via random forest. *Molecules*, 20(6):10947–10962, 2015.

37. J. Ma, R. P. Sheridan, A. Liaw, G. E. Dahl, and V. Svetnik. Deep neural nets as a method for quantitative structure–activity relationships. *Journal of chemical information and modeling*, 55(2):263–274, 2015.

38. V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

39. M. J. O’Meara, S. Ballouz, B. K. Shoichet, and J. Gillis. Ligand similarity complements sequence, physical interaction, and co-expression for gene function prediction. *PloS one*, 11(7):e0160098, 2016.

40. T. Oprea and J. Mestres. Drug repurposing: far beyond new targets for old drugs. *The AAPS journal*, 14(4):759–763, 2012.
41. H. Öztürk, E. Ozkirimli, and A. Özgür. A comparative study of smiles-based compound similarity functions for drug-target interaction prediction. *BMC bioinformatics*, 17(1):128, 2016.

42. T. Pahikkala, A. Airola, S. Pietilä, S. Shakyawar, A. Szwalja, J. Tang, and T. Aittokallio. Toward more realistic drug–target interaction predictions. *Briefings in bioinformatics*, page bbu010, 2014.

43. M. Ragoza, J. Hochuli, E. Idrobo, J. Sunseri, and D. R. Koes. Protein–ligand scoring with convolutional neural networks. *J. Chem. Inf. Model*, 57(4):942–957, 2017.

44. P. W. Rose, A. Prlić, A. Altunkaya, C. Bi, A. R. Bradley, C. H. Christie, L. D. Costanzo, J. M. Duarte, S. Dutta, Z. Feng, et al. The rcsb protein data bank: integrative view of protein, gene and 3d structural information. *Nucleic acids research*, page gkw1000, 2016.

45. P. A. Shar, W. Tao, S. Gao, C. Huang, B. Li, W. Zhang, M. Shahen, C. Zheng, Y. Bai, and Y. Wang. Pred-binding: large-scale protein–ligand binding affinity prediction. *Journal of enzyme inhibition and medicinal chemistry*, 31(6):1443–1450, 2016.

46. K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

47. T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *Journal of molecular biology*, 147(1):195–197, 1981.

48. N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

49. K. Tian, M. Shao, S. Zhou, and J. Guan. Boosting compound–protein interaction prediction by deep learning. In *Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on*, pages 29–34. IEEE, 2015.

50. T. van Laarhoven, S. B. Nabuurs, and E. Marchiori. Gaussian interaction profile kernels for predicting drug–target interaction. *Bioinformatics*, 2011.

51. I. Wallach, M. Dzamba, and A. Heifets. Atomnet: a deep convolutional neural network for bioactivity prediction in structure-based drug discovery. *arXiv preprint arXiv:1510.02855*, 2015.

52. L. Wang, Z.-H. You, X. Chen, S.-X. Xia, F. Liu, X. Yan, Y. Zhou, and K.-J. Song. A computational-based method for predicting drug–target interactions by using stacked autoencoder deep neural network. *Journal of Computational Biology*, 2017.
53. M. Wen, Z. Zhang, S. Niu, H. Sha, R. Yang, Y. Yun, and H. Lu. Deep-learning-based drug–target interaction prediction. *Journal of Proteome Research*, 16(4):1401–1409, 2017.

54. H. Y. Xiong, B. Alipanahi, L. J. Lee, H. Bretschneider, D. Merico, R. K. Yuen, Y. Hua, S. Gueroussov, H. S. Najafabadi, T. R. Hughes, et al. The human splicing code reveals new insights into the genetic determinants of disease. *Science*, 347(6218):1254806, 2015.

55. Y. Yamanishi, M. Araki, A. Gutteridge, W. Honda, and M. Kanehisa. Prediction of drug–target interaction networks from the integration of chemical and genomic spaces. *Bioinformatics*, 24(13):i232–i240, 2008.