GraphDTA: prediction of drug–target binding affinity using graph convolutional networks

Thin Nguyen[1], Hang Le[2], Svetha Venkatesh[1]
[1]Applied Artificial Intelligence Institute, Deakin University, Australia;
[2]Faculty of Information Technology, Nha Trang University, Viet Nam

While the development of new drugs is costly, time consuming, and often accompanied with safety issues, drug repurposing, where old drugs with established safety are used for medical conditions other than originally developed, is an attractive alternative. Then, how the old drugs work on new targets becomes a crucial part of drug repurposing and gains much of interest. Several statistical and machine learning models have been proposed to estimate drug–target binding affinity and deep learning approaches have been shown to be among state-of-the-art methods. However, drugs and targets in these models were commonly represented in 1D strings, regardless the fact that molecules are by nature formed by the chemical bonding of atoms. In this work, we propose **GraphDTA** to capture the structural information of drugs, possibly enhancing the predictive power of the affinity. In particular, unlike competing methods, drugs are represented as graphs and graph convolutional networks are used to learn drug–target binding affinity. We trial our method on two benchmark drug–target binding affinity datasets and compare the performance with state-of-the-art models in the field. The results show that our proposed method can not only predict the affinity better than non-deep learning models, but also outperform competing deep learning approaches. This demonstrates the practical advantages of graph-based representation for molecules in providing accurate prediction of drug–target binding affinity. The application may also include any recommendation systems where either or both of the user- and product-like sides can be represented in graphs.

Availability and implementation The proposed models are implemented in Python. Related data, pre-trained models, and source code are publicly available at https://github.com/thinng/GraphDTA.

1 Introduction

It costs 2.6 billion US dollars to develop a *de novo* drug [22] and takes about 10–17 years for the drug to be accepted/rejected by US FDA [1, 29]. Repurposing/repositioning a drug – identifying new use for an existing approved drug [33] – would reduce the time and cost as several phases spent in the development for its original indication can be

bypassed [1]. Then, how the old drugs work on new targets – drug-target interaction – becomes an important task for drug repurposing and gains much of interest.

Conventionally, high-throughput screening experiments are used to examine bioactivities between drugs and targets, which is a costly and time-consuming procedure [6, 24]. This is impracticable as there are millions of drug-like compounds [10] and hundreds of potential targets, e.g., 500 protein kinases [20], responsible for the modifications of about 30% of human proteins [32]. Thus, it is an important alternative to use statistical and machine learning models to estimate the strength of the interactions for novel couples of drug-target, based on the interactions already measured in clinical experiments.

Several computational approaches/machine learning methods have been proposed for the purpose [16, 15, 7]. In that trend, deep learning models are among the best performers in the prediction of drug–target binding affinity [25]. However in these state-of-the-art deep learning models, for computing convenience, drugs are represented as strings, which is not a natural representation of compounds, and thus the structural information of molecules is lost.

In this paper we propose **GraphDTA** to predict the drug–target binding affinity. In the model drugs are represented as graphs where the edges are the bonding of atoms. Then layers of convolutions on graphs of drugs, concatenated with 1D convolution for protein sequences, were together regressed with the affinity of drug-target couples. We test the proposed method on two benchmark datasets and compare the performance with state-of-the-art approaches in the field. The results show that our proposed method gains the best performance in the task of drug–target binding affinity prediction.

2 Related work

2.1 Drug representation

To represent molecules to be readable by computers, SMILES (Simplified Molecular Input Line Entry System) was invented [38], enabling several efficient applications, including fast retrieval and substructure searching. From SMILES codes, drug descriptors, such as the number of heavy atoms or valence electrons, can be inferred, readily to be used as the features to predict the affinity. One can also consider SMILES codes as strings and use natural language processing (NLP) techniques to featurize the drugs. Alternatively, these strings can be seen as in 1D representation, input into a convolution neural network (CNN) to learn a model to predict the affinity. Layers of 1D convolutions and pooling are used to capture hidden patterns in the inputs, which potentially become powerful predictive features of the affinity.

SMILES as graphs SMILES codes can be also be converted back to graphs, and then geometric deep learning techniques can be applied in downstream analysis.

The most popular deep learning approaches to graphs, graph convolutional networks (GCNs), can be adopted for the task of drug–target binding affinity prediction. GCNs is a generalization of convolution neural networks (CNN) to graph-structured data. GCNs can be divided into two main categories: spectral-based and spatial-based

approaches. In the spectral-based approach, graphs are first represented in spectral domain, then convolutional operation is defined in that domain. Spectral-based GCNs were first proposed in [2], extended by [9, 17, 14].

On the contrary, the spatial-based methods perform learning algorithms directly on graph spatial domain. The learning process includes a neighborhood formulation, then a node information is updated by aggregating information from its neighbor nodes, called message passing, followed by a sub-sampling task. Compared to the spectral-based methods which handle the whole graph simultaneously, the spatial approaches can instead process graph nodes in batches thus can be scalable to large graphs. Recent works on this approach include [21, 23, 12, 37, 36, 40].

GCNs have been used in computational drug discovery [34], including quantitative structure activity/property relationship prediction, interaction prediction, synthesis prediction, and *de novo* molecular design. The problem we explore in this paper, prediction of drug–target binding affinity, belongs to the task of interaction prediction, where the interactions could be among drugs, among proteins, or between drugs and proteins. Examples include Decagon [41], where graph convolutions were used to embed the multi-modal graphs of multiple drugs to predict side effects of drug combinations; or AttSemiGAE/AttTransGAE [19], where attentive multi-view graph auto-encoders were used to measure drug similarity. There are also other approaches in dealing with the problem this work aims to solve, described as follows.

2.2 Prediction of drug–target binding affinity

2.2.1 Affinity similarity (SimBoost)

Apparently the task of drug–target binding affinity prediction could be considered as a collaborative filtering problem (CF). For example, in movie ratings as in the Nexflix competition[1], the rating for a couple of movie-user is learned, or collaboratively filtered, from the ratings by the movies/users similar to the given movie/user. The lesson from Nexflix competition is that if the number of training user-movie ratings is big enough, external information for users or movies does not make significantly contribution to the recommendation systems. However this is not always the case for drug-target binding prediction problem, where the affinity (as ratings in CF) available is often sparse.

The affinity available in training is also used, accompanied with similarities among drugs as well as among targets, to build features in [13] (**SimBoost**). The features are then are the input to gradient boosting machines to predict the binding affinity for unknown drug–target couples.

2.2.2 Kernel based (KronRLS)

Alternatively, the similarity could be built from other sources rather than the affinity in training data. For example, in kernel based approaches, as in [5, 4], kernels for drugs and targets are built from their molecular descriptors, input into a regularized least squares regression model (RLS) to predict the binding affinity.

[1] https://www.netflixprize.com/rules.html

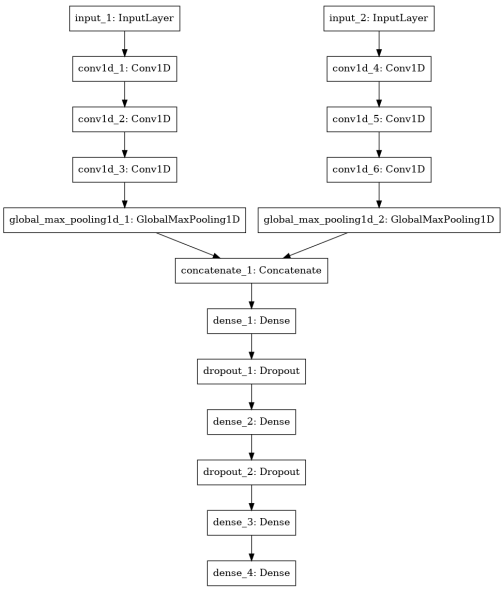

Figure 1: A deep learning model, with drug and targets as strings, to predict the affinity.

Given the problem is to predict the affinity for *n* drugs and *m* targets, there would be *n***m* combinations of them and the kernel would be in the size of $(n * m)^2$. To speed up model training, Cichonska et al. [5, 4] (**KronRLS**) suggest to use KronRLS (Kronecker regularized least-squares). In KronRLS, a pairwise kernel *K* is computed as the Kronecker product of compound kernel of size *n***n* and protein kernel of size *m***m*.

To compute the kernels, any similarity measures can be used. For example, in [27], the kernels for drugs was built from Tanimoto-based similarity (*Pubchem-Sim*)[2]; whilst for targets, *Smith-Waterman* score [31] was used as the similarity measure of protein sequences.

2.2.3 Deep learning (DeepDTA & WideDTA)

When 1D representation for drugs (SMILES) and proteins (sequences) is provided, deep learning could be a possible approach to predict the affinity [25] (**DeepDTA**), as shown in Figure 1. In the figure, *input_1* and *input_2* are drugs and targets, respectively. As these are in 1D representation, layers of 1D convolutions and pooling are used to capture potential patterns in the inputs. They are then concatenated, sent through regularized layers of Dropout, and finally regressed with the training affinity.

WideDTA [26] is an extension of DeepDTA [25] where drugs and proteins are represented as words, instead of characters as in DeepDTA. In particular, drugs are described via most common substructures, denoted as Ligand Maximum Common Substructures (LMCS) [39]; and proteins are represented through most conserved subse-

[2]Provided at http://pubchem.ncbi.nlm.nih.gov

quences, which are Protein Domain profiles or Motifs (PDM), retrieved from PROSITE database [30].

While WideDTA [26] and DeepDTA [25] learned latent feature vectors for the proteins, PADME [11] proposed to used fixed-rule descriptors to represent proteins, such as Protein Sequence Composition (PSC) descriptors [3]. Though, PADME was reported [11] to have similar performance with DeepDTA [25].

3 Proposed method (GraphDTA)

Motivated by bringing nature representation for drugs into modeling of drug-target interaction, we propose a novel deep learning model, **GraphDTA**, for drug-target affinity prediction. Then the DTA prediction problem is cast as a regression task where the input is a pair of protein and drug representations and the output is a continuous value reflecting the affinity binding score of the pair. In existing methods, the input proteins and drugs are treated as sequence representations. Specifically, drugs are represented as SMILES strings – describing the chemical structure in short ASCII strings; and similarly, protein sequences is represented as a string of ASCII letters, which are the amino acids. Having the inputs as strings of text, one conventional approach is to apply various 1D CNN layers to learn latent features on those sequences, similar to natural language processing techniques. Our approach is different, we instead investigate the use the representation of input compounds in the form of graphs, which are capable of capturing bonds among the atoms.

3.1 Graph representation of drug compounds

The drug compounds can be described as graphs of interactions between atoms. Therefore, handling input compounds in the form of graph representation, and subsequently applying learning algorithms on graphs may fit well the task. To this end, for each input compound string (SMILES), we construct a corresponding molecular graph reflecting interactions between the atoms inside the compound. The graph construction and atom feature extraction process are conducted using the RDKit – an open-source chemical informatics software [18].

To describe a node in the graph, we adopt an atom feature design from DeepChem [28][3]. In details, the node feature vector is constituted of five types of atom features: atom symbol, atom degree – number of bonded neighbors plus number of Hydrogen, total number of Hydrogen, implicit value of atom, and whether the atom is aromatic. These atom properties constitute a multi-dimensional binary feature vector.

An edge is set to a pair of atoms if there exists a bond between them. As a results, an indirect, binary graph with attributed nodes is built for each input SMILES string.

3.2 Deep learning on molecular graphs

Having the drug compounds represented in the form of graphs, the problem is to employ an algorithm that enables learning effectively from graph structured data. Recent

[3]https://github.com/deepchem/deepchem

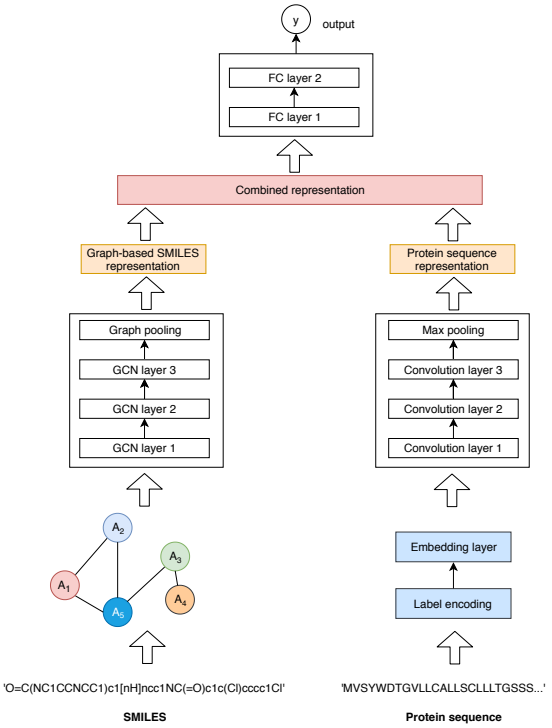

Figure 2: GraphDTA model illustration. Compound SMILES is first converted to a molecular graph, then a deep learning algorithm is adopted on the graph to learn graph representation. Protein sequence is first encoded and embedded, then several 1D CNN layers are applied to learn sequence representation. The two representation vectors are then concatenated and undergo various fully connected layer, ended by a regression layer to estimate the output as the drug-target affinity value.

success of deep convolutional neural networks in computer vision, speech recognition, and natural language processing has led to the idea of extending the convolution operation to graph structures. A number of works have been proposed to handle two main challenges of generalizing CNN to graphs, that are, the formation of receptive fields in graphs whose data points are not arranged as Euclidean grids, and the pooling operation to down-sample a graph.

In this work, we propose a new DTA prediction model based on a combination of graph neural network and conventional CNN. As illustrated in Figure 2, our model takes two inputs: SMILES sequences and protein sequences and feed-forwards them in parallel to learn a representation vector for each, then the two latent feature vectors are concatenated and undergo several dense layers, ended by a regression layer to estimate the affinity value.

For protein sequence input, similar to existing methods, our models takes the sequence as a string of ASCII chars and apply several 1D CNN layers over the text to learn a sequence representation vector. More specifically, the protein sequence is first categorically encoded, then an embedding layer is added to the sequence, each (encoded) character is represented by a 128-dimentional vector. Next, three 1D convolutional layers are used to learn different levels of abstract features from the input. Finally, a max pooling layer is applied to get a representation vector of the input protein sequence.

On the other hand, our model takes the SMILES input as a graph which has been constructed as previously described, then applies a graph convolutional algorithm to learn a representation vector. In order to evaluate the effectiveness of graph-based methods, we investigate several graph neural network models, including GCN [17], GAT [37], GIN [40], and a combined GAT-GCN architecture. The details of each GCN architecture are described as follows.

3.2.1 GCN-based graph representation learning

GCN model [17] was originally designed for the problem of semi-supervised node classification. The model enables to learn hidden layer representations that capture both local graph structures and features of nodes. This fits well our constructed indirect, node attributed graphs. Formally, denote a graph for a given drug as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} \in \mathbb{R}^{N \times F}$ is the set of N nodes each represented by a F-dimentional vector and $\mathcal{E}$ is the set of edges represented as an adjacency matrix $A \in \mathbb{R}^{N \times N}$. The GCN layer is defined by [17] as

$$Z = \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X \Theta \tag{1}$$

where $Z \in \mathbb{R}^{N \times F'}$ is the convolved feature matrix, $\tilde{A}$ is the graph adjacency matrix with added self loop, $\tilde{D}$ is the graph diagonal degree matrix, $X \in \mathbb{R}^{N \times F}$is the node feature matrix, and $\Theta \in \mathbb{R}^{N \times C}$ is the trainable parameter matrix. Note that F and F' are the feature dimensions of input and output nodes, respectively.

To make the GCN applicable to the task of learning a representation vector of the whole graph, we add a global max pooling layer right after the last GCN layer. In our GCN-based model, we use three consecutive GCN layers each activated by a ReLU

function. Then a global max pooling layer is added to aggregate the whole graph representation.

3.2.2 GAT-based graph representation learning

We adopt graph attention network (**GAT**) [37] in our model. Unlike graph convolution techniques, this method proposes an attention-based architecture to learn hidden representations of nodes in a graph by applying a self-attention mechanism. The building block of a GAT architecture is a *graph attention layer.* The GAT layer takes the set of nodes of a graph, the node feature vector $\mathbf{x} \in \mathbb{R}^F$, as input, applies a linear transformation to every node by a weigh matrix $\mathbf{W} \in \mathbb{R}^{F^{'} \times F}$ where F and F' are the feature dimensions of input and output nodes, respectively. At the input node i in the graph, the *attention coefficients* between i and its first-order neighbors are computed as

$$e_{ij} = a(\mathbf{W}\mathbf{x}_i, \mathbf{W}\mathbf{x}_j) \tag{2}$$

The value of e_{ij} indicates the importance of node j to node i. These *attention coefficients* are then normalized by applying a soft-max function, then used to compute the output features for nodes as

$$\sigma(\sum_{j \in \mathcal{N}(i)} \alpha_{ij} \mathbf{W}\mathbf{x}_j) \tag{3}$$

where $\sigma(.)$ is a non-linear activation function and α_{ij} are the normalized *attention coefficients*.

In our model, the GAT-based graph learning architecture includes two GAT layers, activated by a ReLU function, then followed a global max pooling layer to obtain graph representation vector. In details, for the first GAT layer, *multi-head-attentions* are applied with the number of heads is set to 10, and the number of output features are set identical with the number of input features. The output features of the second GAT is set to 128.

3.2.3 Graph Isomorphism Network

We integrate a recently proposed graph learning method, namely Graph Isomorphism Network (**GIN**) [40]. This model is theoretically proven that it achieves maximum discriminative power among GNNs [40]. Specifically, GIN uses a multi-layer perceptrons (MLP) to update the node features as

$$\mathbf{x}_i^{'} = MLP((1+\epsilon)\mathbf{x}_i + \sum_{j \in \mathcal{N}(i)} \mathbf{x}_i) \tag{4}$$

where ϵ is either a learnable parameter or a fixed scalar, $\mathbf{x} \in \mathbb{R}^F$is the node feature vector, and $\mathcal{N}(i)$ is the set of nodes neighbor to *i*.

In our model, the GIN-based graph neural net consists of five GIN layers, each followed by a batch normalization layer. Finally, a global max pooling layer is added to aggregate a graph representation vector.

3.2.4 GAT-GCN combined graph neural network

We investigate a combination of **GAT-GCN** [37, 17] for learning on graphs in our proposed GraphDTA model. In detail, the graph neural network starts by a GAT layer which takes graphs as input and returns the convolved feature matrix to a GCN layer. Each layer is activated by a ReLU function. The graph representation vector is then computed by concatenating the global max pooling and global mean pooling layers from GCN layer output.

4 Experiments and Results

4.1 Experimental setting

To compare with the state of the art performers DeepDTA [25] and WideDTA [26], we ran our proposed method on the same datasets used in these work. In particular, two datasets were used to evaluate the models:

- **Davis** dataset: contains binding affinities observed for all pairs of 72 drugs and 442 targets, measured by Kd value (kinase dissociation constant) [8]. The affinity value ranges from 5.0 to 10.8.

- **Kiba** dataset: contains binding affinities for 2,116 drugs and 229 targets [35]. The affinity value ranges from 0.0 to 17.2.

For these two datasets, same train/test splits with [25] (DeepDTA) and [26] (WideDTA) were used in the experiments, making the comparison as fair as possible. That is, 80% of data instances were used for training and 20% were for testing the models. For the same purpose, same measures as used in [25, 26] were used to evaluate the performance of all the methods: Mean Square Error (MSE, the smaller the better) and Concordance Index (CI, the larger the better). For DeepDTA [25], WideDTA [26] and all the referenced methods, reported results in relevant papers were shown.

4.2 Results and Discussion

Table 1 presents the performance, in MSE or CI measures, for different approaches to predict the affinity for Davis dataset (Table 1a) and Kiba dataset (Table 1b).

For Davis dataset (Table 1a), the best MSE for baseline is *0.261*, gained by DeepDTA [25], when both drugs and proteins are represented as 1D strings. In comparison, all of our proposed graph convolution models achieved better MSE. The best MSE our methods gained is **0.229**, a reduction of **12.3%** over the best baselines.

Meanwhile, on CI measure, there is a slight improvement by the proposed methods. In details, the best CI a baselines could achieve is *0.886*, by DeepDTA [25], when drugs are represented as 1D strings and proteins in Smith-Waterman, and by WideDTA [26], when the both parties are presented as 'words'. In comparison, two graph-based methods, GAT and GIN outperformed the best baselines in CI measures, at **0.892** and **0.893**, respectively.

Similar performance is observed for Kiba dataset (Table 1b). In particular, WideDTA [26] is the best baseline in both measures, CI, at *0.875*, and MSE, at *0.179*, when both drugs and proteins are represented as 'words'. In comparison, three of our

Method	Protein rep.	Compound rep.	CI	MSE
Baseline models				
DeepDTA	Smith-Waterman	Pubchem-Sim	0.790	0.608
DeepDTA	Smith-Waterman	1D	*0.886*	0.420
DeepDTA	1D	Pubchem-Sim	0.835	0.419
KronRLS	Smith-Waterman	Pubchem-Sim	0.871	0.379
SimBoost	Smith-Waterman	Pubchem-Sim	0.872	0.282
DeepDTA	1D	1D	0.878	*0.261*
WideDTA	1D + PDM	1D + LMCS	*0.886*	0.262
Proposed model - **GraphDTA**				
GCN [17]	1D	Graph	0.880	**0.254**
GAT_GCN	1D	Graph	0.881	**0.245**
GAT [37]	1D	Graph	**0.892**	**0.232**
GIN [40]	1D	Graph	**0.893**	**0.229**

(a) For Davis dataset, sorted by MSE. Italics: best for baseline models, bold: better than baselines.

Method	Protein rep.	Compound rep.	CI	MSE
Baseline models				
DeepDTA	1D	Pubchem-Sim	0.718	0.571
DeepDTA	Smith-Waterman	Pubchem-Sim	0.710	0.502
KronRLS	Smith-Waterman	Pubchem-Sim	0.782	0.411
SimBoost	Smith-Waterman	Pubchem-Sim	0.836	0.222
DeepDTA	Smith-Waterman	1D	0.854	0.204
DeepDTA	1D	1D	0.863	0.194
WideDTA	1D + PDM	1D + LMCS	*0.875*	*0.179*
Proposed model - **GraphDTA**				
GAT [37]	1D	Graph	0.866	0.179
GIN [40]	1D	Graph	**0.882**	**0.147**
GCN [17]	1D	Graph	**0.889**	**0.139**
GAT_GCN	1D	Graph	**0.891**	**0.139**

(b) For Kiba dataset, sorted by MSE. Italics: best for baseline models, bold: better than baselines.

Table 1: Prediction performance. Baseline results are from [25]. For our proposed method, same settings as with the referenced methods, e.g., train/test splits, was used.

four proposed models outperformed the best baseline in both of MSE and CI measures. Noticeably, the best MSE the proposed methods gained is **0.139**, a reduction of **22.3%** over the best baseline. Whilst, on CI measure, the proposed methods achieved **0.891**, versus *0.875* by the best baseline, which is not a substantial gain.

Of all the graph convolution models experimented, GIN [40] outperformed the best baselines in both the datasets and both the performance measures. This shows the potential of GIN in graph discrimination/representation, partly supporting the claim in [40] that Graph Isomorphism Network (GIN) is among most powerful GNNs.

5 Summary

In this work, we propose a novel method for estimating drug–target binding affinity, called GraphDTA, which represents drugs as graphs. Using deep convolution networks on graphs of drugs, we show that GraphDTA can not only predict the affinity of drugs–targets better than non-deep learning models, but also outperform competing deep learning methods. In particular, GraphDTA perform consistently well across two separate benchmark databases in all the evaluation measures. The result suggests that representing molecules as graphs improves the performance considerably. Also, it confirms that deep learning models are appropriate for drug–target binding affinity problems. In addition to tackling drug–target binding affinity problems, our methods could be applied to any collaborating filtering problems / recommendations systems where either or both of user- or product-like sides can be represented in graph structures.

Competing interests The authors declare that they have no competing interests.

References

[1] Ted T Ashburn and Karl B Thor. Drug repositioning: identifying and developing new uses for existing drugs. *Nature Reviews Drug Discovery*, 3(8):673, 2004.

[2] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2014.

[3] Dong-Sheng Cao, Qing-Song Xu, and Yi-Zeng Liang. propy: a tool to generate various modes of Chou's PseAAC. *Bioinformatics*, 29(7):960–962, 2013.

[4] Anna Cichonska, Tapio Pahikkala, Sandor Szedmak, Heli Julkunen, Antti Airola, Markus Heinonen, Tero Aittokallio, and Juho Rousu. Learning with multiple pairwise kernels for drug bioactivity prediction. *Bioinformatics*, 34(13):i509–i518, 2018.

[5] Anna Cichonska, Balaguru Ravikumar, Elina Parri, Sanna Timonen, Tapio Pahikkala, Antti Airola, Krister Wennerberg, Juho Rousu, and Tero Aittokallio.

Computational-experimental approach to drug-target interaction mapping: a case study on kinase inhibitors. *PLoS Computational Biology*, 13(8):e1005678, 2017.

[6] Philip Cohen. Protein kinases–the major drug targets of the twenty-first century? *Nature Reviews Drug Discovery*, 1(4):309, 2002.

[7] Steven M Corsello, Joshua A Bittker, Zihan Liu, Joshua Gould, Patrick McCarren, Jodi E Hirschman, Stephen E Johnston, Anita Vrcic, Bang Wong, Mariya Khan, et al. The Drug Repurposing Hub: a next-generation drug library and information resource. *Nature Medicine*, 23(4):405, 2017.

[8] Mindy I Davis, Jeremy P Hunt, Sanna Herrgard, Pietro Ciceri, Lisa M Wodicka, Gabriel Pallares, Michael Hocker, Daniel K Treiber, and Patrick P Zarrinkar. Comprehensive analysis of kinase inhibitor selectivity. *Nature Biotechnology*, 29(11):1046, 2011.

[9] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, pages 3844–3852, 2016.

[10] Mukund Deshpande, Michihiro Kuramochi, Nikil Wale, and George Karypis. Frequent substructure-based approaches for classifying chemical compounds. *IEEE Transactions on Knowledge and Data Engineering*, 17(8):1036–1050, 2005.

[11] Qingyuan Feng, Evgenia Dueva, Artem Cherkasov, and Martin Ester. PADME: A deep learning-based framework for drug-target interaction prediction. *arXiv*, (arXiv:1807.09741), 2018.

[12] Hongyang Gao, Zhengyang Wang, and Shuiwang Ji. Large-scale learnable graph convolutional networks. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*, pages 1416–1424. ACM, 2018.

[13] Tong He, Marten Heidemeyer, Fuqiang Ban, Artem Cherkasov, and Martin Ester. SimBoost: a read-across approach for predicting drug–target binding affinities using gradient boosting machines. *Journal of Cheminformatics*, 9(1):24, 2017.

[14] Mikael Henaff, Joan Bruna, and Yann LeCun. Deep convolutional networks on graph-structured data. *arXiv*, (arXiv:1506.05163), 2015.

[15] Murat Iskar, Georg Zeller, Xing-Ming Zhao, Vera van Noort, and Peer Bork. Drug discovery in the age of systems biology: the rise of computational approaches for data integration. *Current Opinion in Biotechnology*, 23(4):609–616, 2012.

[16] Sarah L Kinnings, Nina Liu, Peter J Tonge, Richard M Jackson, Lei Xie, and Philip E Bourne. A machine learning-based method to improve docking scoring functions and its application to drug repurposing. *Journal of Chemical Information and Modeling*, 51(2):408–419, 2011.

[17] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

[18] Greg Landrum. RDKit: Open-source cheminformatics.

[19] Tengfei Ma, Cao Xiao, Jiayu Zhou, and Fei Wang. Drug similarity integration through attentive multi-view graph auto-encoders. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3477–3483. AAAI Press, 2018.

[20] Gerard Manning, David B Whyte, Ricardo Martinez, Tony Hunter, and Sucha Sudarsanam. The protein kinase complement of the human genome. *Science*, 298(5600):1912–1934, 2002.

[21] Federico Monti, Davide Boscaini, Jonathan Masci, Emanuele Rodola, Jan Svoboda, and Michael M Bronstein. Geometric deep learning on graphs and manifolds using mixture model CNNs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5115–5124, 2017.

[22] Asher Mullard. New drugs cost US $2.6 billion to develop. *Nature Reviews Drug Discovery*, 13(12):877, 2014.

[23] Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkov. Learning convolutional neural networks for graphs. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 2014–2023, 2016.

[24] Martin EM Noble, Jane A Endicott, and Louise N Johnson. Protein kinase inhibitors: Insights into drug design from structure. *Science*, 303(5665):1800–1805, 2004.

[25] Hakime Öztürk, Arzucan Özgür, and Elif Ozkirimli. DeepDTA: deep drug–target binding affinity prediction. *Bioinformatics*, 34(17):i821–i829, 2018.

[26] Hakime Öztürk, Elif Ozkirimli, and Arzucan Özgür. WideDTA: prediction of drug-target binding affinity. *arXiv*, (arXiv:1902.04166), 2019.

[27] Tapio Pahikkala, Antti Airola, Sami Pietilä, Sushil Shakyawar, Agnieszka Szwajda, Jing Tang, and Tero Aittokallio. Toward more realistic drug–target interaction predictions. *Briefings in Bioinformatics*, 16(2):325–337, 2014.

[28] Bharath Ramsundar, Peter Eastman, Patrick Walters, and Vijay Pande. *Deep Learning for the Life Sciences: Applying Deep Learning to Genomics, Microscopy, Drug Discovery, and More*. O'Reilly Media, 2019.

[29] Allen D Roses. Pharmacogenetics in drug discovery and development: a translational perspective. *Nature Reviews Drug Discovery*, 7(10):807, 2008.

[30] Christian JA Sigrist, Lorenzo Cerutti, Edouard De Castro, Petra S Langendijk-Genevaux, Virginie Bulliard, Amos Bairoch, and Nicolas Hulo. PROSITE, a protein domain database for functional characterization and annotation. *Nucleic Acids Research*, 38(suppl_1):D161–D166, 2009.

[31] Temple F Smith and Michael S Waterman. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147(1):195–197, 1981.

[32] Shawn J Stachel, John M Sanders, Darrell A Henze, Mike T Rudd, Hua-Poo Su, Yiwei Li, Kausik K Nanda, Melissa S Egbertson, Peter J Manley, Kristen LG Jones, et al. Maximizing diversity from a kinase screen: identification of novel and selective pan-Trk inhibitors for chronic pain. *Journal of Medicinal Chemistry*, 57(13):5800–5816, 2014.

[33] Stephen M Strittmatter. Overcoming drug development bottlenecks with repurposing: old drugs learn new tricks. *Nature Medicine*, 20(6):590, 2014.

[34] Mengying Sun, Sendong Zhao, Coryandar Gilvary, Olivier Elemento, Jiayu Zhou, and Fei Wang. Graph convolutional networks for computational drug development and discovery. *Briefings in Bioinformatics*, 06 2019.

[35] Jing Tang, Agnieszka Szwajda, Sushil Shakyawar, Tao Xu, Petteri Hintsanen, Krister Wennerberg, and Tero Aittokallio. Making sense of large-scale kinase inhibitor bioactivity data sets: a comparative and integrative analysis. *Journal of Chemical Information and Modeling*, 54(3):735–743, 2014.

[36] Kiran K Thekumparampil, Chong Wang, Sewoong Oh, and Li-Jia Li. Attention-based graph neural network for semi-supervised learning. *arXiv*, (arXiv:1803.03735), 2017.

[37] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[38] David Weininger. SMILES, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, 1988.

[39] Michał Woźniak, Agnieszka Wołos, Urszula Modrzyk, Rafał L Górski, Jan Winkowski, Michał Bajczyk, Sara Szymkuć, Bartosz A Grzybowski, and Maciej Eder. Linguistic measures of chemical diversity and the 'keywords' of molecular collections. *Scientific Reports*, 8, 2018.

[40] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

[41] Marinka Zitnik, Monica Agrawal, and Jure Leskovec. Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics*, 34(13):457–466, 2018.