DeepGS: Deep Representation Learning of Graphs and Sequences for Drug-Target Binding Affinity Prediction

Xuan Lin[1] and **Kaiqi Zhao**[2] and **Tong Xiao**[1] and **Zhe Quan**[1] and **Zhi-Jie Wang**[3] and **Philip S. Yu**[4]

Abstract. Accurately predicting drug-target binding affinity (DTA) in silico is a key task in drug discovery. Most of the conventional DTA prediction methods are simulation-based, which rely heavily on domain knowledge or the assumption of having the 3D structure of the targets, which are often difficult to obtain. Meanwhile, traditional machine learning-based methods apply various features and descriptor, and simply depend on the similarities between drug-target pairs. Recently, with the increasing amount of affinity data available and the success of deep representation learning models on various domains, deep learning techniques have been applied to DTA prediction. However, these methods consider either label/one-hot encodings or the topological structure of molecules, without considering the local chemical context of amino acids and SMILES sequences. Motivated by this, we propose a novel end-to-end learning framework, called `DeepGS`, which uses deep neural networks to extract the local chemical context from amino acids and SMILES sequences, as well as the molecular structure from the drugs. To assist the operations on the symbolic data, we propose to use advanced embedding techniques (i.e., Smi2Vec and Prot2Vec) to encode the amino acids and SMILES sequences to a distributed representation. Meanwhile, we suggest a new molecular structure modeling approach that works well under our framework. We have conducted extensive experiments to compare our proposed method with state-of-the-art models including KronRLS, SimBoost, DeepDTA and DeepCPI. Extensive experimental results demonstrate the superiorities and competitiveness of DeepGS.

1 Introduction

Effectively predicting drug-target binding affinity (DTA) is one of the important problems in drug discovery. Drugs (or ligands) [18] are chemical compounds, each of which can be represented by both a molecule graph with atoms as nodes and chemical bonds as edges, and a string obtained from the Simplified Molecular Input Line Entry System (SMILES) [35]. Targets (or proteins) are sequences of amino acids. Binding affinity indicates the strength of the interactions of drug-target pairs. Through binding, drugs can have a positive or negative influence on functions carried out by proteins, affecting the disease conditions [38]. By understanding drug-target binding affinity, it is possible to find out candidate drugs that are able to inhibit the target/protein and benefits many other bioinformatic applications [26, 27, 14, 39]. As a result, DTA prediction has received much attention in recent years [3, 15, 37].

Early approaches for DTA prediction can be roughly classified into two types: (i) simulation-based methods, and (ii) traditional machine learning-based methods. Simulation-based methods rely on domain knowledge [21] or the 3D structure of target/protein [29, 9], which are often difficult to obtain. Meanwhile, traditional machine learning-based methods apply various features [31, 8] and descriptors [7, 13, 6], and simply depend on the similarities between drug-target pairs [34, 40, 19]. Recently, owing to the remarkable success in various machine learning tasks (*e.g.,* image recognition and natural language processing), deep learning-based methods are also exploited for DTA prediction [23]. These methods consider either label/one-hot encodings or the topological structure of molecules, they, however, do not consider the local chemical context of amino acids and SMILES sequences. It is easily understood that the topological structure information provides an overview of how the atoms are connected, while the local chemical context reveals the functionality of the atoms, like the semantic meaning of a word in a sentence. These two types of information complement each other and are both important for DTA prediction. It should be meaningful and interesting to take these two types of information consideration together. To this end, this paper proposes a novel end-to-end learning framework for DTA prediction, namely **Deep** representation learning framework for **G**raphs and **S**equences (DeepGS).

In a nutshell, our framework consists of three major building blocks. One of the major blocks learns low-dimension vector representations for target/protein sequences, using a convolutional neural network (CNN). The other two blocks learn two representations for drugs, by using a graph attention network (GAT) and a bi-directional gate recurrent unit (BiGRU), respectively. Specifically, (**i**) the CNN and BiGRU blocks extract local chemical context information of amino acids in targets and atoms in drugs, respectively. Since the label/one-hot encodings of amino acids and atoms often neglect the context information, and motivated by the idea of Word2Vec [20], we leverage advanced techniques, *Smi2Vec* and *Prot2Vec*, to encode the amino acids and atoms to a distributed representation, before plugging them to CNN and BiGRU. (**ii**) The newly designed GAT-based molecular structure modeling approach extracts the topological features of drugs, by aggregating the representations of r-radius subgraphs. (**iii**) The learned representations for both drugs and targets are then passed to a neural network to predict the binding affinity.

Different from the existing simulation-based methods, our framework needs neither expert knowledge nor 3D structure of the tar-

[1] College of Computer Science and Technology, Hunan University, Changsha, China.

[2] School of Computer Science, University of Auckland, Auckland, New Zealand.

[3] School of Data and Computer Science, Sun Yat-Sen University, Guangzhou, China; and the Guangdong Key Laboratory of Big Data Analysis and Processing, Guangzhou, China; and the National Engineering Laboratory for Big Data Analysis and Applications, Beijing, China

[4] Department of Computer Science, University of Illinois at Chicago, Chicago, USA.

gets, and so it could be more easy-to-use. Additionally, the proposed framework takes advantage of the local chemical context information of atoms/amino acids in drugs/proteins and uses a newly designed molecular structure modeling approach, which differ DeepGS from the existing deep learning models. To summarize, the main contributions of this paper are listed as follows:

- We propose a novel model DeepGS for DTA prediction. To the best of knowledge, this work is the first to consider both local chemical context and topological structure to learn the interaction between drugs and targets.
- We conduct extensive experiments to study the performance of our proposed method, based on both small and large benchmarking datasets. The experimental results demonstrate (i) the promising performance of our proposed model, (ii) considering jointly local chemical context and topological structure is effective, and (iii) the newly designed molecular structure modeling approach works well under our proposed framework.

The rest of the paper is organized as follows. In Section 2, we introduce the proposed method for drug-target binding affinity prediction. In Section 3, we report and analyze the performance of our method. Section 4 reviews the related work. Finally, we conclude the paper in Section 5.

2 The Proposed Method

In this section, we first provide an overview of the proposed DeepGS framework (Section 2.1). Then, we introduce the representation learning for drugs and targets, respectively (Sections 2.2∼2.3). Finally, we discuss the binding affinity prediction with DeepGS (Section 2.4).

2.1 Overview of DeepGS

Figure 1 shows the overview of DeepGS. It takes the symbolic sequences of target/protein and drug, as well as the molecular structure of the drug as the input. It outputs the binding affinity for the drug-target pair. Remind that the central idea of DeepGS is to consider both local chemical context and the molecular structure, by using some embedding techniques (i.e., Smi2Vec and Prot2Vec) to encode the amino acids and atoms to a distributed representation. Therefore, we design DeepGS as a three-step framework for DTA prediction:

1. Encoding symbolic tokens in target/drug sequences;
2. Encoding the whole drug/target sequences and the molecular structure of the drug; and
3. Predicting the binding affinity value based on the encodings of the drug and the target.

Specifically, motivated by Word2Vec [20], in **the first step** we encode the symbols in the sequence of the target/protein and the drug to a distributed representation, by using *Prot2Vec* and *Smi2Vec*, respectively. Then, the sequences can be transformed into matrices, where each row is the representation of a symbol in the sequences. In **the second step** we extract features, from the drug/target matrices and the molecule graph, to encode the whole sequences and graph. For the target/protein, we consider the local chemical context of the amino acids, by using a convolutional neural network (CNN). For the drug, we consider both the molecular structure and the local chemical context. Particularly, since the molecular structure can be represented by a graph, we suggest a graph attention network (GAT)
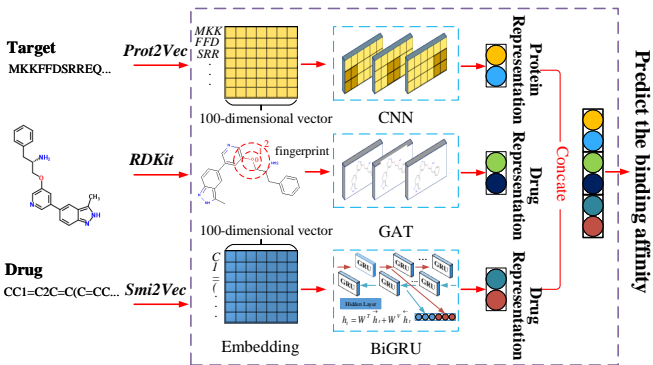

Figure 1. Overview of DeepGS.

based approach to extract the topological information of the drug. In the meantime, the local chemical context of atoms in the drug is captured, by using a bi-directional gated recurrent unit (BiGRU). As a result, we obtain a latent representation for the target and two latent presentations for the drug. To predict the binding affinity, **in the third step** DeepGS inputs the concatenation of the three latent representations to a stack of fully connected layers, and outputs a real value binding affinity. Next, we present the details of our method.

2.2 Representation Learning for Drug

2.2.1 Local Chemical Context Modeling

Drugs are often presented in the format of SMILES (Simplified Molecular-Input Line-Entry System), a specification in the form of a line notation for describing the structure of chemical compound [35]. For example, the SMILES string of the drug in Figure 1 is "CC1=C2C=C(C=CC...", which is a sequence of atoms and covalent bonds. For ease of representation, we consider both atoms and covalent bonds as symbolic tokens, and so the SMILES sequence is a sequence of symbols. To encode the SMILES sequence, existing deep learning approaches such as *DeepDTA* [30] use label/one-hot encoding to represent each symbol in the SMILES sequence. However, label/one-hot encoding often neglects the context of the symbol, and thus cannot reveal the functionality of the symbol within the context. To remedy this, we propose to use *Smi2Vec* [25], a method similar to Word2Vec [20], to represent the symbols in the SMILES sequence. Algorithm 1 shows the pseudo-codes of encoding SMILES symbols, based on the pre-trained *Smi2Vec* embeddings. In general, a SMILES string with fixed length, say m, is divided into a separate atom or symbol (Line 1). Then, it *maps* the atom by looking up each of the atom embeddings from the pre-trained dictionary, while it randomly generates values if it is not in the dictionary (Lines 2-6). Finally, it constructs an atom matrix A by aggregating embedding vectors (Lines 7-8), where each line represents the pre-trained vector of an atom.

Motivated by the gate function in GRU [5], we apply a 1-layer Bi-GRU on the resulting matrix to obtain a latent representation of the drug, which allows us to model the local chemical context. Note that BiGRU takes a fix-sized of matrix as the input, while the length of SMILES strings may vary. One simple solution is to fix the length of input sequence at approximately average length of the SMILES string in the dataset, and apply zero-paddings at the end of the input sequences. As we will show later in Section 3, an appropriate length (e.g., larger than the average length of sequences in the dataset) does

---

Algorithm 1: Smi2Vec

Input: a molecule s in the format of SMILES, dictionary D, atom vector's fixed length m, vector dimension d.

Output: atom matrix A

1   atom set $\{x_j | 1 \leq j < |s|\} \longleftarrow split(s)$
2   **for** $j=1$ to m **do**
3   **if** $x_j \notin dictionary$ **then**
4   embedding vector $a_j \longleftarrow$ randomly generated value $\in \Re^d$
5   **else**
6       $a_j \xleftarrow{map} x_j$ // by using D
7   atom matrix $A \longleftarrow \sum_{j=1}^{m} a_j$
8   **return** $A \in \Re^{m \times d}$

---

not make the performance of our framework change a lot. Considering the training efficiency and the DTA performance, it is suggested that a small number is a good trade-off between efficiency and performance.

2.2.2 *Molecular Structure Modeling*

In addition to the local chemical context, we exploit the molecular structure to uncover how the atoms connect in the drug. The molecular structure is an important cue for DTA prediction [30]. To achieve this, we can first use the RDKit [11] tool to transform SMILES string of a chemical compound into a molecule graph $G = (V, E)$, in which the node $v_i \in V$ represents the i-th atom, and $e_{ij} \in E$ represents the chemical bond between the i-th and the j-th atoms. Then, we can learn a graph attention network (GAT) [32] from the molecule graphs G. To apply GAT on molecule graph, we can encode all atoms and chemical bonds to a d-dimensional vector, and aggregates the information from the r-radius subgraph for each atom in the molecular graph, where r is the number of hops from an atom. Algorithm 2 shows the pseudo-codes of applying GAT on molecule graphs. Specifically, it first computes an initial vector concatenating the fingerprint (i.e., the r-radius subgraph) and the adjacent information for each atom (Lines 1-4). Here it leverages Weisfeiler-Lehman algorithm to extract the fingerprint of the atoms. Then, it updates the atom vectors by propagating the information from its neighboring nodes (Lines 5-6). Finally, it aggregates the atom vectors to obtain the representation of the molecule (Line 7), each of which contains the r-radius subgraph information.

Algorithm 2: GAT on molecule graph

Input: Molecule graph $G = (V, E)$, radius R

Output: a vector $y_{molecule}$ for a molecule

1   **for** *each node* $v_i \in V$ **do**
2     $adj(v_i) \leftarrow extract\_adjacency(G)$
3     $fp(v_i) \leftarrow extract\_fingerprints(v_i, G, R)$
4     $V_{in} \leftarrow [adj(v_i); fp(v_i)]$
5   **for** *each node* $v_i \in V$ **do**
6   update $V_{in} \leftarrow V_{in} + \sum_{v_j \in Neighbors(v_{in})} GATConv(V_j)$
7   **return** $y_{molecule} \leftarrow \sum_{v_1}^{V} V_{in}$

2.3 Representation Learning for Target/Protein

Targets/proteins are often represented as a sequence of amino acids (*e.g.*, MKKFFDSRREQ... shown in Figure 1). Similar to the

SMILES string, we propose to first encode the amino acids into a d-dimensional vector following *Prot2Vec* [2], which allows us to capture local chemical information in targets/proteins. As a single amino acid often makes no sense, we apply a fixed-length N-gram splitting approach to partition the sequence into meaningful "biological words". Note that, here the sequence refers to the *fixed-length input protein sequence* (instead of the full sequences), which is preprocessed as similar as we handle the SMILES strings (recall Section 2.2). Compared to the commonly used label encoding methods, the fixed-length N-gram divides the sequence into a sequence of N-grams. Each N-gram is considered as a "biological word". Intuitively, it can generates more "words context" than label encoded by one-hot encoding.

Considering that there are generally 20 kinds of amino acids, rendering that the maximum number of possible N-grams is 20^N. To make trade-off between the training feasibility and vocabulary size, in our paper we define $N = 3$. Specifically, given a protein or target sequence $L = \{x_i | (i = 1, 2, ..., |l|)\}$, where x_i represents the i-th amino acid and $|l|$ represents the sequence length, the fixed-length 3-gram splitting method partitions the sequence into the following 3-grams, each of which is a biological word consisting of 3 amino acids: $[x_1; x_2; x_3], [x_4; x_5; x_6], ..., [x_{|l|-2}; x_{|l|-1}; x_{|l|}]$. For each biological word, we map it to an embedding vector by looking up a pretrained embedding dictionary for 9048 words [2], which is obtained from Swiss-Prot (https://www.uniprot.org/) with 560,118 manually annotated sequences. As a result, we transform each target sequence to a matrix, in which each row is the embedding of a biological word. The matrix is then fed into a CNN to extract the local chemical context of the target. It is worth noting that, different from the early ligand-based approach [13] that neglects the local context information in targets/proteins, our solution above leverages the embedding technique to learn the representation from the protein sequence.

2.4 Drug-target Binding Affinity Prediction

In this study, we look on drug-target prediction as a regression task by predicting the binding affinity values. With the representation learned from the previous sections, we can integrate all the information from drugs and targets to predict the binding affinity. In brief, we concatenate all the representations and feed them to three fully-connected dense layers to output the affinity value. More precisely, for the GAT block, we use two graph attention layers to update the node vectors in a graph considering their neighbor nodes. For the CNN block, we use three consecutive 2D-convolutional layers. And for the BiGRU block, we use one BiGRU layer. Besides, we use Rectified Linear Unit (ReLU) [22] as the activation function, which has been commonly adopted in deep learning research. Given a set of drug-target pairs and the ground-truth affinity values in the training dataset, we can use the mean square error (MSE) as the loss function:

$$\mathcal{L}_{MSE} = \frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i - y_i)^2$$

where $\hat{y}_i$ is the predicted value, y_i is the ground-truth value, and N represents the number of drug-target pairs.

3 Experiments

In this part, we first describe the experimental settings (Section 3.1). Then, we compare our proposed method with state-of-the art models (Section 3.2). Besides, we conduct more experiments to analyze

our model including the prediction performance and sensitiveness (Section 3.3). Finally, we conduct an ablation study to investigate the effectiveness of main strategies suggested in the paper (Section 3.4).

3.1 Experimental Setup

3.1.1 Datasets

Following prior works [23, 12], we employed widely-used datasets that are specialized for DTA prediction:

- The **Davis** dataset[5], which contains 68 drugs and 442 targets with 30,056 drug-target interactions.
- The **KIBA** dataset[6], which originally comes from a method named Kinase Inhibitor BioActivity (KIBA), and it introduces KIBA scores with integration of the statistic information of K_d, K_i and IC_{50} into a single bioactivity score for drug-target interaction. The dataset contains 2,111 drugs and 229 targets with 118,254 interactions after processing [23].

As same as prior works [23, 12], we split the datasets into 6 subsets with the equal size, and used five of them for training and the remaining one for testing.

3.1.2 Evaluation Metrics

We used four metrics commonly used in regression task (recall Section 2.4) to evaluate the performance. They include: Mean Squared Error (MSE), Concordance Index (CI), r_m^2, and Area Under Precision Recall (AUPR) score.

MSE has been defined in the previous section as the objective of DeepGS. **CI** [23] measures whether the predicted binding affinity values rank the corresponding drug-target interactions in the same order as the ground-truth does. It is computed as

$$CI = \frac{1}{Z} \sum_{y_i > y_j} \zeta(f_i - f_j)$$

$$\zeta(b) = \begin{cases} 1, & \text{if } b > 0 \\ 0.5, & \text{if } b = 0 \\ 0, & \text{if } b < 0 \end{cases},$$

where Z is a normalization constant that equals the number of drug-target pairs with different binding affinity values. More specifically, when $y_i > y_j$, a positive score is given if and only if the predicted $f_i > f_j$. Here, $\zeta(b)$ is a step function.

The metric $\mathbf{r_m^2}$ is used to evaluate the external prediction performance of QSAR (Quantitative Structure-Activity Relationship) models. A model is acceptable if and only if $r_m^2 \geq 0.5$.

$$r_m^2 = r^2 * (1 - \sqrt{r^2 - r_0^2})$$

where r^2 and r_0^2 represent the squared correlation coefficient values between the observed and predicted values with and without intercept, respectively.

The **AUPR** score is widely used for binary classification. A commonly used binding affinity value is defined based on the logarithm of K_d as follow:

$$pK_d = -log10(\frac{K_d}{1e9})$$

where K_d refers to the dissociation value [28]. Here, we transformed the datasets into binary datasets with predefined thresholds. We followed the prior work [23] to select pK_d value of 7 as threshold for the Davis dataset, and pK_d value of 12.1 as threshold for the KIBA dataset.

3.1.3 Baseline Methods

We compared DeepGS against the following state-of-the-art models:

- **KronRLS**[7][24]: This approach is based on Kronecker Regularized Least Square. It aims to minimize the objective function, $J(f) = \sum_{i=1}^m (y_i - f(x_i))^2 + \lambda \| f \|_k^2$, where x_i (i=1,...,m) is a set of training input features, f is a non-linear function, y_i represents their corresponding real-valued labels, and $\lambda > 0$ is a pre-defined regularization parameter, $\| f \|_k^2$ is the norm of f with kernel k.
- **SimBoost [12]:** This baseline constructs three kinds of features and trains a gradient boosting machine [4] model to represent the nonlinear associations between the input features and the binding affinities.
- **DeepCPI [30]:** This baseline is originally designed for CPI/DTI prediction, and cannot be used directly for DTA task. Here, we need to change it to a regression task. Specifically, we replaced the loss function of *cross_entropy* with *MSE*, and set the dimension of out layer to 1. The rest is consistent with the original paper.
- **DeepDTA [23]:** DeepDTA trains two 3-layer CNNs with label/one-hot encodings of compound and protein sequences to predict DTA task. Their CNN model consists of two separate CNN blocks to learn the features from SMILES strings of compounds and protein sequences, respectively. The representations of drugs and targets are concatenated and passed to a fully connected layer for DTA prediction.

3.1.4 Implementation Details

For Smi2Vec, we used an embedding layer with 100 dimensions to represent the symbols in SMILES sequences, and for Prot2Vect we used 100-dimensional pre-trained representations for the biological words. As a result, we constructed matrices with (100, 100) and (2000, 100) dimensions for drug and target, respectively. In our experiments, when the molecular graph was used, we employed the RDKit [11] software to convert the textual representation in SMILES format to a graph representation. For the GAT block, we set the number of heads to 10, and it was implemented using pytorch_geometric[8], and we set the same radius $r = 2$ as in [30]. For the BiGRU block, we set the size of input and hidden layer to 100. For the CNN block, we set the size of kernel to 23. Note that, we performed grid search over a combination of the hyper-parameters to determine the settings. The detailed settings are summarized in Table 1. And we obtained a high performance of the proposed framework with a relatively small range on hyper-parameter tuning. The proposed framework was implemented using PyTorch[9] with Tensorflow [1] backend and ADAM [16] optimization. Our experiments were run on Linux 16.04.10 with Intel(R) Xeon(R) CPU E5-2678 v3@2.50GHz and GeForce GTX 1080Ti (11GB).

---

[5] http://staff.cs.utu.fi/ aatapa/data/DrugTarget/
[6] https://pubs.acs.org.ccindex.cn/doi/suppl/10.1021/ci400709d

[7] http://staff.cs.utu.fi/ aatapa/software/RLScore/
[8] https://github.com/rusty1s/pytorch_geometric
[9] https://github.com/pytorch/pytorch

Table 1. The detailed training settings of DeepGS. The length of SMILES/protein sequence has three various settings, which are used to study the impact of SMILES/protein sequence length.

Parameter	Setting	Parameter	Setting
Radius r	2	Layer of CNN	3
N-gram	3	Layer of BiGRU	1
CNN kernel size	23	Learning rate (lr)	1e-4
Length of SMILES sequence	50, 100, 500	lr decay	0.9
Length of protein sequence	500, 1000, 2000	Decay interval	20
Vector dimension	32	Weight decay	1e-5
Window size	11	Epoch	100
Depth in GAT	2	Batchsize	1

3.2 Comparison Results

To examine the competitiveness of the proposed model, we compared DeepGS with state-of-the-art models (including classic and deep learning models) used for DTA prediction. Table 2 reports the average CI, MSE, r_m^2 and AUPR scores on the Davis dataset.

From this table, we can see that, on the whole classic methods such as KronRLS perform worse than deep learning-based methods. This is because classic methods rely heavily on hand-crafted features and the similarity matrices of drugs and targets. In contrast, deep learning-based approaches capture more information via automatic feature engineering with CNN and GNN. In addition, we find that our method performs better than other two deep learning-based methods. The reason could be that (i) compared to DeepCPI, our method jointly considers topological structures and local chemical context, which is benefit to the performance; (ii) compared to Deep-DTA, we incorporate GAT model to obtain the topological information of drug and advanced embedding techniques which bring more contextual information than one-hot vectors for modeling both drugs and targets.

Overall, this set of experiments demonstrate that our proposed method DeepGS outperforms all these baselines in all metrics. This is a very encouraging result. It is worth noting that, although the improvements seem to be small at the first glance, it is essentially a non-trivial achievement in terms of DTA prediction.

Table 2. The average CI, MSE, r_m^2 and AUPR scores on the Davis dataset. The results of KronRLS, SimBoost and DeepDTA are reported from [23].

Method	Drugs	Targets	CI	MSE	r_m^2	AUPR
KronRLS	Pubchem	S-W	0.871	0.379	0.407	0.661
SimBoost	Pubchem	S-W	0.872	0.282	0.644	0.709
DeepCPI	GNN	CNN (Embedding)	0.867	0.293	0.607	0.705
DeepDTA	CNN	CNN (One-hot)	0.878	0.261	0.630	0.714
DeepGS	GAT+Smi2Vec	CNN (Prot2Vec)	**0.882**	**0.252**	**0.686**	**0.763**

Besides the comparison on the Davis dataset, we also conduct the comparison on the KIBA dataset. Table 3 shows the comparison results. It can be seen that, the overall performance tendency is similar to that on the Davis dataset. For example, the performance of Kron-RLS is inferior to that of deep learning-based approaches, the performance of DeepCPI is inferior to that of DeepDTA, and our method exhibits better performance on almost all these metrics. This further demonstrates the competitiveness of DeepGS. Note that, in terms of CI metric, our method still has the comparable performance to Deep-DTA, since the value of our method is only slightly smaller than that of DeepDTA. The possible reason is that, the KIBA dataset comes from multiple sources (*e.g.*, K_i, K_d and IC_{50}, recall Section 3.1), the data heterogeneity in KIBA dataset may make a negative effect on the CI metric of our model.

Table 3. The average CI, MSE, r_m^2 and AUPR scores on the KIBA dataset. The results of KronRLS, SimBoost and DeepDTA are reported from [23].

Method	Drugs	Targets	CI	MSE	r_m^2	AUPR
KronRLS	Pubchem	S-W	0.782	0.411	0.342	0.635
SimBoost	Pubchem	S-W	0.836	0.222	0.629	0.760
DeepCPI	GNN	CNN (Embedding)	0.852	0.211	0.657	0.782
DeepDTA	CNN	CNN (One-hot)	**0.863**	0.194	0.673	0.788
DeepGS	GAT+Smi2Vec	CNN (Prot2Vec)	0.860	**0.193**	**0.684**	**0.801**

3.3 Model Analysis

In this section, we conduct more experiments to analyze our model. In the first experiment, we further examine the prediction performance of our model based the predicted value (p) and measured value (m). In the second experiment, we examine the sensitiveness of our model by using various sequence lengths.
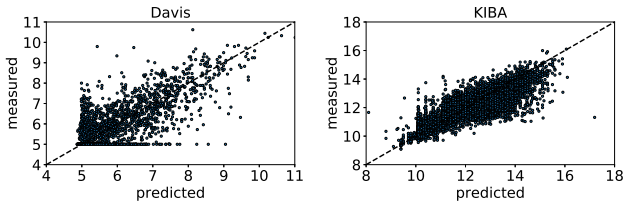
Figure 3. Predictions from DeepGS model against measured binding affinity values. The left and right figures plot the results on the Davis and KIBA datasets, respectively.

Figure 3 plots the predicted value (p) and measured value (m) on these two datasets. Note that, a good model should hold that predicted value (p) is close to the measured value (m), and thus the samples should fall close to the dashed ($p = m$) line. One can see that, for the Davis dataset, the dense area of the pK_d value is in the range of 5 to 6 in terms of x-axis. This is because the pK_d value of 5 constitutes more than half of the dataset (i.e., 20,931 out of 30,056, as reported from [23]). In addition, we observe that the dense area of the KIBA score is in the range of 10 to 14 in terms of x-axis. The
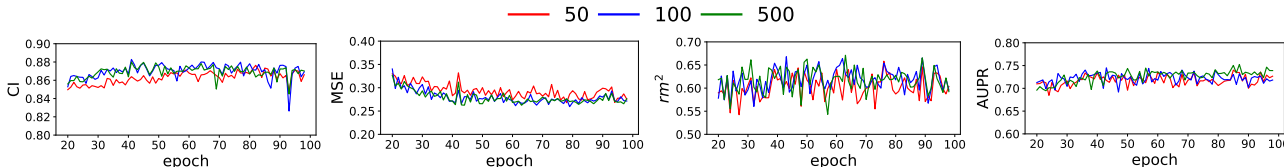
Figure 2. CI, MSE, r_m^2, and AUPR vs. L_{ds} (drug SMILES sequences in the Davis dataset).
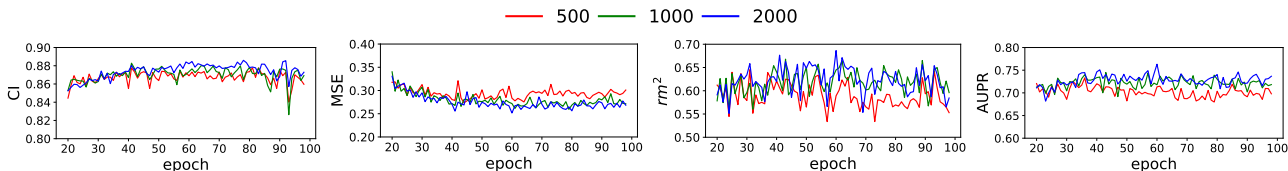
Figure 4. CI, MSE, r_m^2, and AUPR vs. L_{ps} (protein sequences in the Davis dataset).

reason is similar to that for the Davis dataset. Particularly, for both datasets, the samples are close to the dashed ($p = m$) line. This justifies, from another perspective, that the proposed solution has a good prediction performance.
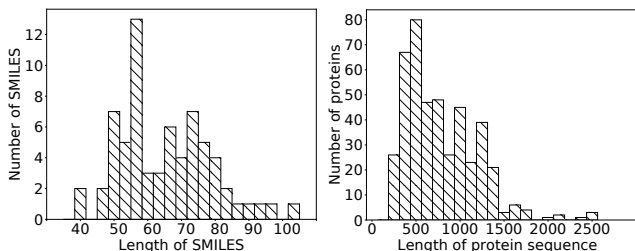

Figure 5. Sequence length distribution in Davis dataset. The left and right figures refer to the length distribution of SMILES and protein sequences, respectively.

To investigate the sensitiveness of our model, a simple way is to remove some information of the input sequences, and then to test the model's prediction performance. In this paper, we use the following scheme that not only can remove some information of input sequences but also can partially reflect the impact of sequence length. Specifically, we fix the length of the input sequences at some value, say l, when training BiGRU and CNN. To achieve this, we cut the input sequence if the length of the input sequence is longer than l, and otherwise we use the zero-padding at the end of the input sequence. As for the drug's input sequence (i.e., SMILES), we set its length to [50, **100**, 500], as shown in Table 1. Note that, the average length of SMILES sequences in the Davis dataset is 64 and the **bold** refers to the default value. Correspondingly, we set the length of target/protein's input sequence (i.e., anino acids) to [500, **1000**, 2000]. Here, the average length of protein sequences in the Davis dataset is 788. For clarity, we use L_{ds} and L_{ps} to denote the Length of drug sequence and that of protein sequence, respectively.

Figure 2 shows the results when we set L_{ds} to 50, 100 and 500, re-

spectively. Meanwhile, Figure 4 shows the results when we set L_{ps} to 500, 1000 and 2000, respectively. We observe that, (i) the performance gap between $L_{ds} = 100$ and $L_{ds} = 500$ is very tiny, and the performance gap between $L_{ps} = 1000$ and $L_{ps} = 2000$ is not so obvious; (ii) the performance gap between $L_{ds} = 50$ and $L_{ds} = 100$ can be easily perceived, and the performance gap between $L_{ps} = 500$ and $L_{ps} = 1000$ is obvious. This phenomenon is a little bit strange at the first glance. To dig out the reason behind it, we plot the distribution of sequence lengths, as shown in Figure 5. It can be seen that the lengths of most SMILES sequences are less than 100 and larger than 50. Thus, it is natural that the performance gap between $L_{ds} = 100$ and $L_{ds} = 500$ is very tiny, since almost all SMILES sequences do not need to be cut even if $L_{ds} = 100$, i.e., few information is missing. However, when $L_{ds} = 50$, many SMILES sequences may need to be cut, and so the performance degrades. With the similar argument, it is not hard to understand that the performance gap between $L_{ps} = 500$ and $L_{ps} = 1000$, since most protein sequences need to be cut when $L_{ps} = 500$. The reason for the relatively small performance gap between $L_{ps} = 1000$ and $L_{ps} = 2000$ can be inferred with the similar argument. This result may imply that when the sequence length l is set to a value larger than the average length of sequences in the dataset, the performance degradation could be trivial.

3.4 Ablation Study

As mentioned before, existing models for DTA prediction have leveraged the topological structure to learn the representation for drug and target/protein, while they often ignored the local chemical context. Thereby, this work considers both local chemical context and topological structure to learn the interaction between drugs and targets. More precisely, the core idea of DeepGS is to fully leverage local chemical context, by using advanced embedding techniques, to better learn the drug and target representations. To study the effectiveness of the central idea, we implemented a variant of our model, called
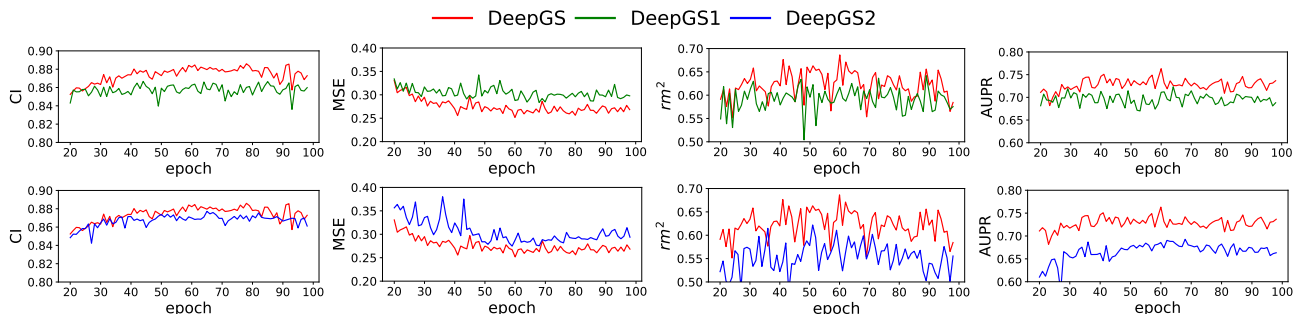

Figure 6. Ablation study on all metrics for our proposed model and two variants on the Davis dataset.

DeepGS1. This variant model removes both drug's and protein's local chemical context information obtained by Smi2Vec and Prot2Vec from the framework. The detailed configuration of DeepGS1 is illustrated in Table 4. In the following experiments, we use the same experimental settings mentioned in Table 1.

Figure 6 shows the comparison results. It can be seen that DeepGS1 is basically inferior to DeepGS in terms of CI, MSE, rm^2 and AUPR. These results demonstrate that combining the local chemical context information, which reflects the functionalities of the atoms, is benefit to learning a good representation for drugs and proteins, improving the prediction performance.

Table 4. The detailed description of the variant of our model. DeepGS1 is mainly for investigating the local chemical context information, while DeepGS2 is used to justify the choice of GAT used in our molecular structure modeling component.

Model	Drug Representation	Target Representation
DeepGS	FP+GAT & Smi2Vec+BiGRU	Prot2Vec+CNN
DeepGS1	FP+GAT & one-hot/Label+BiGRU	one-hot/Label+CNN
DeepGS2	FP+GNN & Smi2Vec+BiGRU	Prot2Vec+CNN

In addition to examining the effectiveness of the central idea, we also conduct another experiment, which is used to answer the following interesting question. Recall Section 2.2.2, we develop a GAT-based molecular structure modeling approach. One could argue that, GNN (Graph Neural Network) can also map a graph to a vector that encodes the topological structure of the graph, since recent work [30] have showed that GNN can effectively model drugs. To address it, we implemented another variant of our model, called DeepGS2. Compared to our model, the major difference is that it uses a GNN-based molecular structure modeling approach (cf., Table 4).

The experimental results are also shown in Figure 6. We can see that, the variant DeepGS2 is obviously poorer than our model. These results justify our choice in Section 2.2.2. The reasons could be two-fold: (i) the molecular structure may contribute a lot to the prediction performance; and (ii) The changes to the molecular structure modeling approach are sensitive to the model, especially when local chemical context is also considered in the model.

4   Related Work

Drug-target binding prediction has been an interesting topic in drug discovery. Most of previous works focused on simulation-based methods (i.e., molecule docking and descriptors) or machine learning-based models. For example, Li et al. [17] proposed a docking method based on random forrest (RF). The RF model was also adopted in *KronRLS* [24] with a similarity score through the Kronecker product of similarity matrix to improve the predictive performance. To remedy the limitation of linear dependencies in KronRLS, a gradient boosting method was proposed in *SimBoost* [12] to construct the similarities among drugs and targets. Although classic methods show reasonable performance in drug-target prediction, they are often computational expensive, or require external expert knowledge or the 3D structure of target/protein, which are difficult to obtain. Different from the classic methods, the proposed framework is able to automatically extract features from the data, and requires neither expert knowledge nor 3D structure of the target/protein. These salient features make the proposed framework applicable to large scale affinity data which is becoming available.

Owing to the great success of deep learning, much attention has been devoted to applying deep learning techniques for drug-target prediction. Most of the existing methods are based on topological similarity. For example, in [36] they developed a Deep Belief Network (DBN) model constructed by stacking Restricted Boltzmann Machines (RBMs). Instead of using DBN, a nonlinear end-to-end learning model named NeoDTI [33] was proposed. NeoDTI integrates variety of information from heterogeneous network data and uses topology-preserving based representations of drugs and targets to facilitate drug-target prediction. With the increasing popularity of graph neural networks (GNNs), researchers are adopting GNNs model for drug prediction. For example, graph convolutional network was used to model molecule based on the extraction of their circular fingerprint [6]. By learning from molecular structures and protein sequences, Gao et al. [10] proposed a neural model for drug-target prediction and used a two-way attention mechanism to provide biological interpretation of the prediction. Ma et al. [19] proposed to use multi-view graph auto-encoders to obtain better inter-pretability and they also added attentive mechanism to determine the weights for each view, according to the corresponding tasks and features. Moreover, Zitnik et al. [41] presented a Decagon model used for modeling polypharmacy side effects, their model constructs a multimodal graph of various interactions (i.e., protein-protein interactions, drug-target interactions) and the polypharmacy side effects.

Among the research on deep learning for drug discovery, DeepDTA [23] and DeepCPI [30] are the most relevant to our work. Both of them addressed the problem of drug-target prediction. DeepDTA takes label/one-hot encodings of compound/protein sequences as input, and trains two CNNs for the drug and target, respectively, to predict the binding affinity value of drug-target pairs. DeepCPI is originally specialized for DTI prediction, it uses a traditional GNN based on representation of r-radius fingerprints to encode the molecular structure of drugs, and a CNN to encode protein sequences. Attention mechanism is adopted to concatenate drug and protein representations for prediction. Here, we take measures to revise it to be used for DTA prediction. Compared with DeepDTA and DeepCPI, the proposed framework considers both local chemical context and the topological information of drugs at the same time to improve the binding affinity prediction, by using Smi2Vec and Prot2Vec to encode the atoms in drugs and amino acids in targets, while the existing methods consider only one of these important factors. Moreover, our work also suggests a new molecular structure modeling approach that works well under our framework.

5   Conclusions

Accurately predicting DTA is a vital and challenging task in drug discovery. In this paper, we have proposed an end-to-end deep learning framework named DeepGS for DTA prediction. It combines a GAT model to extract the topological information of molecular graph and a BiGRU model to obtain the local chemical context of drug. To assist the operations on the symbolic data, we used advanced embedding techniques (i.e., Smi2Vec and Prot2Vec) to encode the amino acids and SMILES sequences to a distributed representation. We have conducted extensive experiments to compare our proposed method with state-of-the-art models. The experimental results demonstrate that the promising performance of our proposed method. This study opens several future research directions: 1) investigating whether our method can be further improved by integrating other state-of-the-art techniques, for example, Generative Adversarial Networks; 2) extending our method to other types of problems in data mining and bioinformatics fields.

REFERENCES

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al., 'Tensorflow: A system for large-scale machine learning', in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pp. 265–283, (2016).

[2] Ehsaneddin Asgari and Mohammad RK Mofrad, 'Continuous distributed representation of biological sequences for deep proteomics and genomics', *PloS one*, **10**(11), e0141287, (2015).

[3] Monica Campillos, Michael Kuhn, Anne-Claude Gavin, Lars Juhl Jensen, and Peer Bork, 'Drug target identification using side-effect similarity', *Science*, **321**(5886), 263–266, (2008).

[4] Tianqi Chen and Tong He, 'Higgs boson discovery with boosted trees', in *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 69–80, (2015).

[5] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio, 'Gated feedback recurrent neural networks', in *International Conference on Machine Learning*, pp. 2067–2075, (2015).

[6] David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams, 'Convolutional networks on graphs for learning molecular fingerprints', in *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 2224–2232, (2015).

[7] Jean-Loup Faulon, Milind Misra, Shawn Martin, Ken Sale, and Rajat Sapra, 'Genome scale enzyme–metabolite and drug–target interaction predictions using the signature molecular descriptor', *Bioinformatics*, **24**(2), 225–233, (2007).

[8] Ronen Feldman, Oded Netzer, Aviv Peretz, and Binyamin Rosenfeld, 'Utilizing text mining on online medical forums to predict label change due to adverse drug reactions', in *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1779–1788. ACM, (2015).

[9] Alex Fout, Jonathon Byrd, Basir Shariat, and Asa Ben-Hur, 'Protein interface prediction using graph convolutional networks', in *NeurIPS*, pp. 6530–6539, (2017).

[10] Kyle Yingkai Gao, Achille Fokoue, Heng Luo, Arun Iyengar, Sanjoy Dey, and Ping Zhang, 'Interpretable drug target prediction using deep neural representation.', in *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 3371–3377, (2018).

[11] L. Greg et al. Rdkit: Open-source cheminformatics, 2006.

[12] 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).

[13] Laurent Jacob and Jean-Philippe Vert, 'Protein-ligand interaction prediction: an improved chemogenomics approach', *Bioinformatics*, **24**(19), 2149–2156, (2008).

[14] Bo Jin, Haoyu Yang, Cao Xiao, Ping Zhang, Xiaopeng Wei, and Fei Wang, 'Multitask dyadic prediction and its application in prediction of adverse drug-drug interaction', in *Thirty-First AAAI Conference on Artificial Intelligence*, (2017).

[15] Michael J Keiser, Vincent Setola, John J Irwin, Christian Laggner, Atheir I Abbas, Sandra J Hufeisen, Niels H Jensen, Michael B Kuijer, Roberto C Matos, Thuy B Tran, et al., 'Predicting new molecular targets for known drugs', *Nature*, **462**(7270), 175, (2009).

[16] Diederik P. Kingma and Jimmy Ba, 'Adam: A method for stochastic optimization', in *ICLR*, (2015).

[17] Hongjian Li, Kwong-Sak Leung, Man-Hon Wong, and Pedro Ballester, 'Low-quality structural and interaction data improves binding affinity prediction via random forest', *Molecules*, **20**(6), 10947–10962, (2015).

[18] Ruoyu Li, Sheng Wang, Feiyun Zhu, and Junzhou Huang, 'Adaptive graph convolutional neural networks', in *Thirty-Second AAAI Conference on Artificial Intelligence*, (2018).

[19] Tengfei Ma, Cao Xiao, Jiayu Zhou, and Fei Wang, 'Drug similarity integration through attentive multi-view graph auto-encoders', in *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 3477–3483, (2018).

[20] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, 'Distributed representations of words and phrases and their compositionality', in *Advances in neural information processing systems (NeurIPS)*, pp. 3111–3119, (2013).

[21] Subhabrata Mukherjee, Gerhard Weikum, and Cristian Danescu-Niculescu-Mizil, 'People on drugs: credibility of user statements in health communities', in *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 65–74, (2014).

[22] Vinod Nair and Geoffrey E Hinton, 'Rectified linear units improve restricted boltzmann machines', in *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 807–814, (2010).

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

[24] 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).

[25] Zhe Quan, Xuan Lin, Zhi-Jie Wang, Yan Liu, Fan Wang, and Kenli Li, 'A system for learning atoms based on long short-term memory recurrent neural networks', in *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 728–733. IEEE, (2018).

[26] Didier Rognan, 'Chemogenomic approaches to rational drug design', *British journal of pharmacology*, **152**(1), 38–52, (2007).

[27] Víctor Suárez-Paniagua and Isabel Segura-Bedmar, 'Using recursive neural networks to detect and classify drug-drug interactions from biomedical texts', in *ECAI*, pp. 1666–1667. IOS Press, (2016).

[28] 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).

[29] Oleg Trott and Arthur J Olson, 'Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading', *Journal of computational chemistry*, **31**(2), 455–461, (2010).

[30] Masashi Tsubaki, Kentaro Tomii, and Jun Sese, 'Compound–protein interaction prediction with end-to-end learning of neural networks for graphs and sequences', *Bioinformatics*, **35**(2), 309–318, (2018).

[31] Twan van Laarhoven, Sander B Nabuurs, and Elena Marchiori, 'Gaussian interaction profile kernels for predicting drug–target interaction', *Bioinformatics*, **27**(21), 3036–3043, (2011).

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

[33] Fangping Wan, Lixiang Hong, An Xiao, Tao Jiang, and Jianyang Zeng, 'Neodti: neural integration of neighbor information from a heterogeneous network for discovering new drug–target interactions', *Bioinformatics*, **35**(1), 104–111, (2018).

[34] Kejian Wang, Jiazhi Sun, Shufeng Zhou, Chunling Wan, Shengying Qin, Can Li, Lin He, and Lun Yang, 'Prediction of drug-target interactions for drug repositioning only based on genomic expression similarity', *PLoS computational biology*, **9**(11), e1003315, (2013).

[35] 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).

[36] Ming Wen, Zhimin Zhang, Shaoyu Niu, Haozhi Sha, Ruihan Yang, Yonghuan Yun, and Hongmei Lu, 'Deep-learning-based drug-target interaction prediction', *Journal of proteome research*, **16**(4), 1401–1409, (2017).

[37] Cao Xiao, Ping Zhang, W. Chaovalitwongse, Jianying Hu, and Fei Wang, 'Adverse drug reaction prediction with symbolic latent dirichlet allocation', in *Thirty-First AAAI Conference on Artificial Intelligence*, pp. 1590–1596, (2017).

[38] Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec, 'Graph convolutional policy network for goal-directed molecular graph generation', in *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 6410–6421, (2018).

[39] Shuangjia Zheng, Yongjian Li, Sheng Chen, Jun Xu, and Yuedong Yang, 'Predicting drug protein interaction using quasi-visual question answering system', *bioRxiv*, 588178, (2019).

[40] Xiaodong Zheng, Hao Ding, Hiroshi Mamitsuka, and Shanfeng Zhu, 'Collaborative matrix factorization with multiple similarities for predicting drug-target interactions', in *SIGKDD*, pp. 1025–1033. ACM, (2013).

[41] Marinka Zitnik, Monica Agrawal, and Jure Leskovec, 'Modeling polypharmacy side effects with graph convolutional networks', *Bioinformatics*, **34**(13), i457–i466, (2018).