Bioinformatics, 38(9), 2022, 2571–2578
https://doi.org/10.1093/bioinformatics/btac155
Advance Access Publication Date: 11 March 2022
Original Paper

Systems biology

BridgeDPI: a novel Graph Neural Network for predicting drug–protein interactions

Yifan Wu[1,2,†], Min Gao[1,†], Min Zeng[2], Jie Zhang [1,3,4,*] and Min Li [2,*]

[1]SenseTime Research, Shanghai 200233, China, [2]School of Computer Science and Engineering, Central South University, Changsha 410083, China, [3]Qing Yuan Research Institute, Shanghai Jiao Tong University, Shanghai 200240, China and [4]Merck Advisory Committee for AI-enabled Health Solution, Shanghai 200126, China

*To whom correspondence should be addressed.
[†]The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.
Associate Editor: Jinbo Xu

Received on August 19, 2021; revised on January 20, 2022; editorial decision on March 6, 2022; accepted on March 10, 2022

Abstract

Motivation: Exploring drug–protein interactions (DPIs) provides a rapid and precise approach to assist in laboratory experiments for discovering new drugs. Network-based methods usually utilize a drug–protein association network and predict DPIs by the information of its associated proteins or drugs, called 'guilt-by-association' principle. However, the 'guilt-by-association' principle is not always true because sometimes similar proteins cannot interact with similar drugs. Recently, learning-based methods learn molecule properties underlying DPIs by utilizing existing databases of characterized interactions but neglect the network-level information.

Results: We propose a novel method, namely BridgeDPI. We devise a class of virtual nodes to bridge the gap between drugs and proteins and construct a learnable drug–protein association network. The network is optimized based on the supervised signals from the downstream task—the DPI prediction. Through information passing on this drug–protein association network, a Graph Neural Network can capture the network-level information among diverse drugs and proteins. By combining the network-level information and the learning-based method, BridgeDPI achieves significant improvement in three real-world DPI datasets. Moreover, the case study further verifies the effectiveness and reliability of BridgeDPI.

Availability and implementation: The source code of BridgeDPI can be accessed at https://github.com/SenseTime-Knowledge-Mining/BridgeDPI. The source data used in this study is available on the https://github.com/IBM/InterpretableDTIP (for the BindingDB dataset), https://github.com/masashitsubaki/CPI_prediction (for the C.ELEGANS and HUMAN) datasets, http://dude.docking.org/ (for the DUD-E dataset), repectively.

Contact: limin@mail.csu.edu.cn or zhangjie1@sensetime.com

1 Introduction

Developing a new drug conventionally takes tens of years and billions of dollars (Avorn, 2015; Paul *et al.*, 2010). In the process of new drug development, drug–protein interaction (DPI) prediction is a crucial step. Traditional experimental assays for determining DTIs are to measure the value of half-maximal inhibitory concentration (IC50) or inhibitory constant (Ki) in wet experiments. Although the experimental assays remain the most reliable approach for examining DPIs, they are time-consuming and cost-intensive due to the need of individual experiments for all of the possible drug–protein pairs. Therefore, exploring the interaction mechanism between drugs and proteins and developing some efficient computational methods for predicting DPIs are very significant and urgently demanded.

The interaction mechanism between drugs and proteins is very complicated, so it is a big challenge to develop an efficient computational method that can accurately determine DPIs. Current computational methods for predicting DPIs are classified into three categories: docking-based methods, network-based methods and learning-based methods. Docking-based methods usually use molecular dynamic simulation to reconstruct the contact relationship between proteins and drugs in space. These methods aim to look for the best binding position inside the binding pocket of the proteins for drug molecules (Gschwend *et al.*, 1996; Led and Caflisch, 2018). Nevertheless, they require accurate 3D structures of proteins, which are hard to obtain while some proteins even do not have the 3D structure (Liu and Altman, 2015; Mizianty *et al.*, 2014; Zhang *et al.*, 2012). In contrast to the docking-based methods, network-based methods bypass the direct reconstructing

their contact relationship by using the 'guilt-by-association' principle to predict DPIs based on some priori DPI data (Ballester and Mitchell, 2010; Bleakley and Yamanishi, 2009; Ding *et al.*, 2014; Durrant and McCammon, 2011; Luo *et al.*, 2017, 2019). The 'guilt-by-association' principle assumes that if a target has a similar profile with another target, the former target is likely to interact with a drug, which can directly interact with the latter target (Luo *et al.*, 2021; Wang and Lukasz, 2019). Thus, these methods usually need to construct a network that contains the existing drugs and proteins and calculate similarity scores of the drug–drug and protein–protein pairs. However, the 'guilt-by-association' principle relies on the quality of the similarity scores, and cannot be easily applied to low-frequency or unseen proteins. Moreover, the 'guilt-by-association' principle is not always true, because there are also some similar proteins that cannot interact with similar drugs (Maggiora *et al.*, 2014). Recently, with the plentiful accumulation of data, learning-based methods have been successfully applied to predict DPIs (Li *et al.*, 2020, 2022; Wan *et al.*, 2019; Wang *et al.*, 2021; Wang and Zeng, 2013; Yuvaraj *et al.*, 2021). Learning-based methods usually take protein sequences and drug molecules as the input for DPI prediction. Compared to the network-based methods, learning-based methods mainly focus on learning the interaction mechanism of each single drug–protein pair by the prior data but ignore some network-level information, i.e. the 'guilt-by-association' principle, which is quite crucial to infer DPIs. Therefore, it is very important and necessary to design a model that brings the 'guilt-by-association' principle into the learning-based methods.

In this paper, we develop BridgeDPI, a deep learning framework, which simultaneously combines the advantages of network-based methods and learning-based methods, to predict DTIs. Compared to previous learning-based methods, we introduce protein sequences and drug molecules into a supervised drug–protein association network. The network provides neighborhood information of proteins and drugs, which makes the model not only learn the interaction mechanism of drug–protein pairs but also provides a network-level perspective to assist learning. In the process of constructing the supervised network, we introduce a class of nodes called bridge nodes. The bridge nodes are devised to connect all the proteins or drugs and measure the associations among proteins/drugs from a network-level perspective. From the network-based point of view, through the bridge nodes, we can get two types of paths from a protein to a drug: explicit paths and implicit paths. As shown in Figure 1, considering the protein–drug pair P1–D1, we can go from P1 to bridge nodes to D1, where the bridge nodes explicitly measure the interaction between P1 and D1 and thus the interaction mechanism is learned to decide whether the pair interacts. Paths like P1 to bridge nodes to D1 are defined as the explicit paths. From another point of view, we can also go from P1 to bridge nodes to P2 to bridge nodes to D2 to bridge nodes to D1, where the bridge nodes not only measure the associations between proteins (i.e. P1–P2)/drugs (i.e. D2–D1) but also measure the interactions between proteins and drugs (i.e. P2–D2). In this way, the interaction between P1 and D1 can be implicitly inferred by P1–P2, P2–D2, and D2–D1. Paths like this type are the implicit paths. In a word, due to the bridge nodes, BridgeDPI not only learns a deep interaction mechanism but also assists DPI prediction from a network-level perspective. This makes BridgeDPI more comprehensively grasp features to perform DPI prediction and the prediction results are also more reliable.

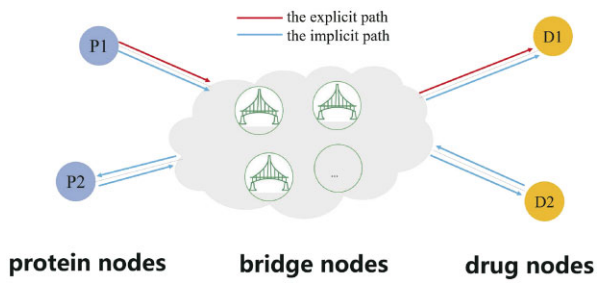

Fig. 1. A toy example of our constructed graph. The graph includes three types of nodes: protein nodes, drug nodes, bridge nodes; three types of edges: bridge node-protein node, bridge node-drug node, bridge node-bridge node; and two types of paths: the explicit path (e.g. P1->bridge nodes->D1) and the implicit path (e.g. P1->bridge nodes->P2->bridge nodes->D2->bridge nodes->D1)

We provide comprehensive comparison results with other baselines in four different datasets. Compared to the state-of-the-art (SOTA) results, BridgeDPI achieves the area under the receiver operating characteristic curve (AUROC) scores of 97.5% (1.9% higher) in the BindingDB dataset, 99.5% (0.7% higher) in the C.ELEGANS dataset and 99.0% (1.1% higher) in the HUMAN dataset. In addition, we use the Directory of Useful Decoys, Enhanced (DUD-E) dataset as an independent test set to evaluate the generalization. BridgeDPI trained on the BindingDB dataset gets the AUROC scores of 77.2% (3.2% higher). In summary, all results indicate that BridgeDPI is effective and reliable in predicting DPIs.

2 Materials and methods

2.1 Datasets

BindingDB: The BindingDB dataset has collected affinity data of 2 286 319 drug–protein pairs from corresponding research papers, where 8536 proteins and 989 383 drugs are included (Gilson *et al.*, 2016; the raw BindingDB can be accessed at https://www.bindingdb.org/bind/index.jsp, and the Gao's version of BindingDB can be downloaded from https://github.com/IBM/InterpretableDTIP). On this basis, Gao *et al.* (2018) select the data having IC50 value and convert the IC50 values into 1 for interactions (IC50 < 100 nM), 0 for no interactions (IC50 > 10 000 nM) to construct a binary classification dataset. The dataset contains 39 747 positive samples and 31 218 negative samples and is divided into training (28 240 positive and 21 915 negative samples), validation (2831 positive and 2776 negative samples) and test (2706 positive and 2802 samples) sets. We use this dataset for our main head-to-head comparisons.

C.ELEGANS and HUMAN datasets: C.ELEGANS and HUMAN datasets have been widely used in DPI prediction (the balanced versions of C.ELEGANS and HUMAN datasets can be downloaded from https://github.com/masashitsubaki/CPI_prediction). Both of them are constructed by combining a set of highly credible negative drug–protein samples via an *in silico* screening method with the known positive samples (Liu *et al.*, 2015). We follow Tsubaki *et al.* (2019) and use the balanced versions to do research. The C.ELEGANS dataset has 7786 drug–protein pairs, including 1876 proteins and 1767 drugs. The HUMAN dataset has 6728 drug–protein pairs, including 2001 proteins and 2726 drugs. Both the C.ELEGANS dataset and the HUMAN dataset are randomly divided for 5-fold cross-validation.

DUD-E dataset: The DUD-E is a widely used dataset covering 102 proteins and 22 886 clustered ligands (Mysinger *et al.*, 2012; the DUD-E dataset can be accessed at http://dude.docking.org/). There are 50 decoys for each active with similar physical and chemical properties but dissimilar 2D topology. It contains 1 429 790 protein-ligand samples in total (22 645 positive samples, 1 407 145 negative samples). The samples are demonstrated by wet experiments or computational methods. DUD-E is used as an independent test set to evaluate how our model performs in reality. In this paper, we train BridgeDPI on the BindingDB dataset and test it on the DUD-E dataset.

2.2 Framework of BridgeDPI

We propose a novel end-to-end deep learning framework, namely BridgeDPI, for DPI prediction task. The overall learning architecture is illustrated in Figure 2. BridgeDPI takes protein sequences and drug SMILES (Weininger, 1988) as inputs and predicts their interactions. It consists of four parts: drug feature extraction part,

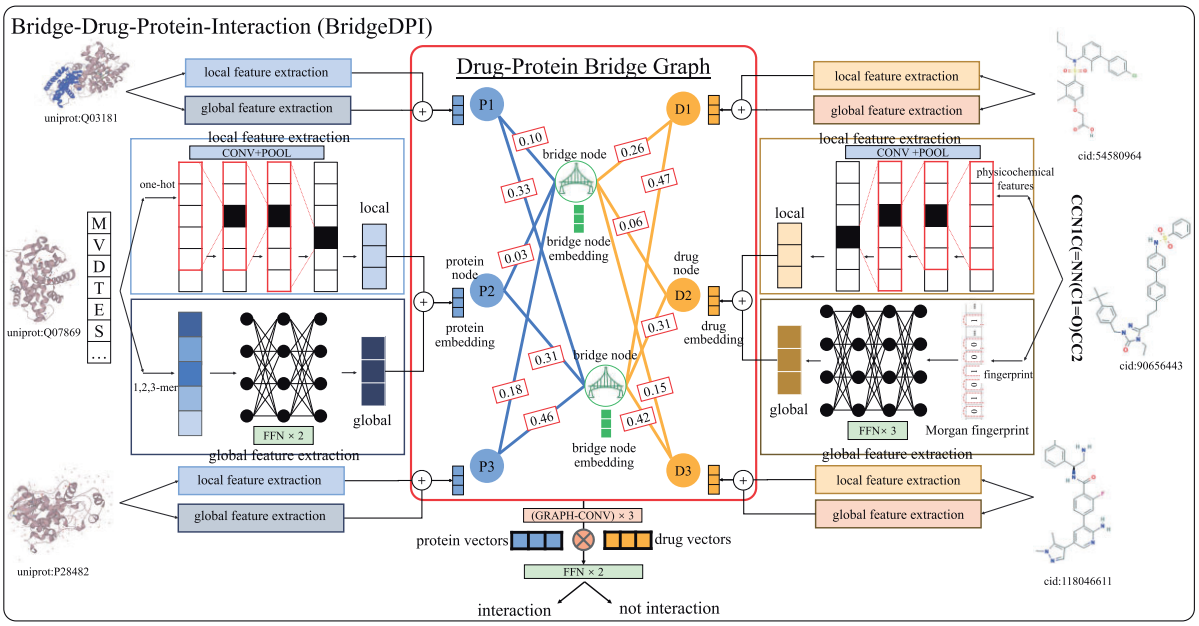

Fig. 2. The framework of BridgeDPI: on the left is the protein feature extraction part, on the right is the drug feature extraction part, the middle is the drug–protein bridge graph construction part, and the bottom is the classification part. BridgeDPI uses CNN and FFN to extract proteins'/drugs' local and global features. After that, some bridge nodes are introduced to construct the bridges between proteins and drugs and thus we can use a GNN to obtain the graph embeddings of proteins and drugs. Finally, we feed the graph embedding into a linear layer with sigmoid activation to predict DPIs

protein feature extraction part, drug–protein bridge graph construction part and classification part. For the drug and protein feature extraction parts, the Convolutional Neural Network (CNN) layers and some Feed-Forward Network (FFN) layers are applied to extract the features from the drug SMILES and the protein sequences. For the drug–protein bridge graph construction part, some bridge nodes are introduced to construct the bridges between proteins and drugs, and thus we can use a Graph Neural Network (GNN) to capture the network-level information to predict DPIs. For the classification part, we get element-wise product of the proteins' and drugs' graph embeddings after GNN, and then feed it into a linear layer with sigmoid activation to predict the interactions. In this section, we will describe the details of each component in BridgeDPI.

2.2.1 Feature extraction of proteins

Before feeding proteins into BridgeDPI, we need to describe them as numeric vectors. In order to better describe the properties of protein sequences, we vectorize them from both a local and global perspective. For the local view, we employ some CNN filters to capture the key local patterns in the protein sequences. Firstly, the one-hot encoding is used to encode the protein's primary amino acid sequences. Then, a 1D CNN with max-pooling is applied to extract the local features. Finally, we transform the CNN's output by a two-layer FFN to get the final local features of the protein. For the global view, we select the protein's k-mer statistics as its global features because the k-mer information reveals the distribution of global characteristics and measures biological similarity for discrimination (Leslie *et al.*, 2004). In our study, we set $k=1$, 2, 3, which generate 20- ($k=1$), 400- (20×20, $k=2$) and 8000- ($20\times20\times20$, $k=3$) dimension vectors, respectively. The vectors are normalized and concatenated to represent the protein from a global view. Here, we give up $k\geq 4$ because it generates too many ($\geq 160\,000$) dimensions and makes the method too complex, easily leading to overfitting and time-consuming training. Specifically, for protein i with length l_p, its final representation $P_i\in\mathbb{R}^{d_h}$ is vectorized as:

$$P_i = P_i^{\text{local}} + P_i^{\text{global}}, \tag{1}$$

where $P_i^{\text{local}}\in\mathbb{R}^{d_h}$ and $P_i^{\text{global}}\in\mathbb{R}^{d_h}$ are protein i's local and global features, d_h is the dimension of them. The P_i^{local} and P_i^{global} are computed by Equation (2), $p_i\in\mathbb{R}^{l_p\times 20}$ is the one-hot embedding of the protein i, $a_i\in\mathbb{R}^{8420}$ is the concatenated vectors of the 1-mer, 2-mer, 3-mer statistic vectors.

$$\begin{aligned} P_i^{\text{local}} &= \text{FFN}_p^{\text{local}}(\text{CNN}_p(p_i)) \\ P_i^{\text{global}} &= \text{FFN}_p^{\text{global}}(a_i) \end{aligned}, \tag{2}$$

where $\text{FFN}_p^{\text{local}}(\cdot)$ is a one-layer FFN, $\text{FFN}_p^{\text{global}}(\cdot)$ is a two-layer FFN, $\text{CNN}_p(\cdot)$ is the 1D CNN with max-pooling. And the output of them is all activated by the Rectified Linear Unit (ReLU) function.

2.2.2 Feature extraction of drugs

After representing proteins, drug molecules also need to be vectorized. Analogously, we also extract the local view's and global view's features from drug molecules. Different from the protein sequences, drug molecules are graphs with atoms as nodes and chemical bonds as edges. It means that the statistics of k-mer-like information are not appropriate anymore. Therefore, we choose another global view's representation technique: molecular fingerprint. The molecular fingerprint encodes a drug molecular into a series of binary digits, where some substructure and topological information are implied (Rogers and Hahn, 2010). As for the local view's representation, similar to proteins, we also employ some CNNs. Firstly, we encode each atom of the drug into a 75-dimension vector, which contains physical–chemical features of atoms and bonds (Ramsundar *et al.*, 2019). Next, a CNN with max-pooling is used to extract features from the 75-dimension vectors and the extracted features are fed into a 3-layer FFN to obtain the final local features of the drug molecules. Specifically, for drug j with l_d atoms, its final representation $D_j\in\mathbb{R}^{d_h}$ is defined as:

$$D_j = D_j^{\text{local}} + D_j^{\text{global}}, \tag{3}$$

where $D_j^{\text{local}}\in\mathbb{R}^{d_h}$ and $D_j^{\text{global}}\in\mathbb{R}^{d_h}$ are drug j's local and global features, d_h is the dimension of them which is equal to the dimension

of proteins' final representations. The D_j^{local} and D_j^{global} are calculated by Equation (4), $d_j \in \mathbb{R}^{l_d \times 75}$ is the 75-dimension per-atom features, $b_j \in \mathbb{R}^{1024}$ is the molecular fingerprint of the drug j.

$$\begin{aligned} D_j^{\text{local}} &= \text{FFN}_{\text{d}}^{\text{local}}(\text{CNN}_{\text{d}}(d_j)) \\ D_j^{\text{global}} &= \text{FFN}_{\text{d}}^{\text{global}}(b_j) \end{aligned}, \tag{4}$$

where $\text{FFN}_{\text{d}}^{\text{local}}(\cdot)$ is a one-layer FFN, $\text{FFN}_{\text{d}}^{\text{global}}(\cdot)$ is a three-layer FFN, $\text{CNN}_{\text{d}}(\cdot)$ is the 1D CNN with max-pooling. And the output of them is all activated by the ReLU function.

2.2.3 Bridge graph's construction

After obtaining the final representations of proteins and drugs, the next thing is to introduce the network-level information. To do this, we construct a supervised drug–protein association network, called bridge graph. Specifically, we introduce a class of nodes called bridge nodes to the constructed network to supervisedly measure the associations among proteins/drugs and the interactions between drug–protein pairs. The bridge nodes are actually some d_{h}-dimension vectors in the space of P_i and D_j, and their associations are defined as the cosine similarities between them:

$$\begin{aligned} \alpha_{P_i,B_k} &= \frac{P_i \cdot B_k}{||P_i||_2||B_k||_2} \\ \alpha_{D_j,B_k} &= \frac{D_j \cdot B_k}{||D_j||_2||B_k||_2} \end{aligned}, \tag{5}$$

where $P_i \in \mathbb{R}^{d_{\text{h}}}$ and $D_j \in \mathbb{R}^{d_{\text{h}}}$ are the final representations of protein i and drug j, $B_k \in \mathbb{R}^{d_{\text{h}}}$ represents a bridge node, $\cdot$ is to compute the inner product of vectors, $||\cdot||_2$ is to compute the two-norm value of a vector. The vectors of the bridge nodes are initialized randomly from a normal distribution $\mathcal{N}(0,1)$. In our model, we employ m bridge nodes (i.e. $B_1, B_2, \ldots, B_m$) to jointly measure the associations and the interactions: for the protein–drug pair $(i-j)$, their interaction can be measured by α_{P_i,B_k} and α_{D_j,B_k}; for another protein–drug pair $(u-v)$, their interaction can be also measured in the same way; moreover, if we think about the both pairs $(i-j)$ and $(u-v)$, with the bridge nodes as its medium, the association between protein i and protein u can be inferred by α_{P_i,B_k} and α_{P_u,B_k}, and the association between drug_j and drug_v can be inferred in the same way. In other words, the associations' and the interactions' information, or the network-level information, is embedded in the constructed network with m bridge nodes and protein/drug nodes, and thus we can use a GNN to capture the information.

In detail, for the protein–drug pair $(i-j)$ and the m bridge nodes, we first compute the cosine similarity between them to obtain a weighted adjacency matrix of the graph:

$$A_{(i-j)} = \begin{pmatrix} \alpha_{P_i,P_i} & \alpha_{P_i,D_j} & \alpha_{P_i,B_1} & \ldots & \alpha_{P_i,B_m} \\ \alpha_{D_j,P_i} & \alpha_{D_j,D_j} & \alpha_{D_j,B_1} & \ldots & \alpha_{D_j,B_m} \\ \alpha_{B_1,P_i} & \alpha_{B_1,D_j} & \alpha_{B_1,B_1} & \ldots & \alpha_{B_1,B_m} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \alpha_{B_m,P_i} & \alpha_{B_m,D_j} & \alpha_{B_m,B_1} & \ldots & \alpha_{B_m,B_m} \end{pmatrix}, \tag{6}$$

where $A_{(i-j)} \in \mathbb{R}^{(m+2)\times(m+2)}$ is the weighted adjacency matrix, $\alpha_{\cdot,\cdot}$ is the cosine similarity between nodes defined as Equation (5). For the stability of convergence, we filter out the negative edges and do normalization for $A_{(i-j)}$:

$$L_{(i-j)} = D^{-\frac{1}{2}}\text{ReLU}(A_{(i-j)})D^{-\frac{1}{2}}, \tag{7}$$

where $D^{-\frac{1}{2}} \in \mathbb{R}^{(m+2)\times(m+2)}$ is the degree matrix of $A_{(i-j)}$, $\text{ReLU}(\cdot)$ is to filter out the negative edges.

Then, to capture the network-level information, a 3-layer GNN is implemented as:

$$\begin{aligned} Z^0 &= \begin{pmatrix} P_i \\ D_i \\ B_1 \\ \ldots \\ B_m \end{pmatrix} \\ Z^1 &= \text{ReLU}(L_{(i-j)}Z^0W^1 + b^1) \\ Z^2 &= \text{ReLU}(L_{(i-j)}Z^1W^2 + b^2) \\ Z^3 &= \text{ReLU}(L_{(i-j)}Z^2W^3 + b^3) \end{aligned}, \tag{8}$$

where $Z^0 \in \mathbb{R}^{(m+2)\times d_{\text{h}}}$ is the initial node embedding of the graph, $Z^1, Z^2, Z^3 \in \mathbb{R}^{(m+2)\times d_{\text{h}}}$ are the node embeddings after aggregating the neighbor information, $W^1, W^2, W^3 \in \mathbb{R}^{d_{\text{h}}\times d_{\text{h}}}$ and $b^1, b^2, b^3 \in \mathbb{R}^{1\times d_{\text{h}}}$ are the hidden parameters of the GNN, $\text{ReLU}(\cdot)$ is the activation function.

Finally, we select the first two rows of Z^0, Z^1, Z^2, Z^3 and sum them up to get the final vectors of protein i and drug j which have aggregated the network-level information.

$$\begin{aligned} \hat{P}_i &= \sum_{k=0}^{3} Z_1^k \\ \hat{D}_j &= \sum_{k=0}^{3} Z_2^k \end{aligned}, \tag{9}$$

where $Z_1^k \in \mathbb{R}^{1\times d_{\text{h}}}$ is the first row of Z^k, $Z_2^k \in \mathbb{R}^{1\times d_{\text{h}}}$ is the second row of Z^k, $\hat{P}_i, \hat{D}_j \in \mathbb{R}^{1\times d_{\text{h}}}$ are the final vectors of protein i and drug j.

2.2.4 Classification

After aggregating the network-level information to the protein and drug representations, the last thing is to infer whether the drug–protein pair interacts. We use element-wise product of drugs' and proteins' final vectors to model the interaction mechanism and after a two-layer FFN the interaction probabilities are predicted:

$$\hat{y}_{(i-j)} = \text{FFN}^{\text{output}}(\hat{P}_i \odot \hat{D}_j), \tag{10}$$

where $\odot$ is to compute the element-wise product of two vectors, $\text{FFN}^{\text{output}}$ is a two-layer FFN with ReLU activation for its first layer and Sigmoid activation for its second layer, $\hat{y}_{(i-j)} \in \mathbb{R}$ is the predicted interaction probability. In order to make the predicted interaction probabilities close to the true interaction values, we use a binary cross entropy loss function as our training objective, and the L2 regularization is added to improve the model's robustness.

$$\begin{aligned} \mathcal{L} = & \frac{1}{|\mathcal{P}|}\sum_{(i-j)\in\mathcal{P}}[y_{(i-j)}\log(\hat{y}_{(i-j)}) + (1-y_{(i-j)})log(1-\hat{y}_{(i-j)}))] \\ & +\lambda\sum_{\theta\in\Theta}\|\theta\|^2 \end{aligned}, \tag{11}$$

where $\mathcal{P}$ is the set of all protein–drug pairs in the training set, $y_{(i-j)} \in \mathbb{R}$ is the true interaction value for protein–drug pair $(i-j)$,

Table 1. The setting of hyper-parameters in BridgeDPI

Hyper-parameter		Value
Protein	Filter size of CNN_{p}	25
	Filter num of CNN_{p}	64
	Neuron num of $\text{FFN}_{\text{p}}^{\text{local}}$	128
	Neuron num of $\text{FFN}_{\text{p}}^{\text{global}}$	1024, 128
Drug	Filter size of CNN_{d}	7
	Filter num of CNN_{d}	64
	Neuron num of $\text{FFN}_{\text{d}}^{\text{local}}$	128
	Neuron num of $\text{FFN}_{\text{d}}^{\text{global}}$	1024, 256, 128
Bridge graph	Num of bridge nodes	64
	Neuron num of GNN	128, 128, 128
Classification	Neuron num of $\text{FFN}^{\text{output}}$	128, 1

Table 2. Performances of BridgeDPI and baselines on BindingDB dataset

Methods	Overall test set			Seen protein set			Unseen protein set		
	ACC (%)	AUROC (%)	AUPR (%)	ACC (%)	AUROC (%)	AUPR (%)	ACC (%)	AUROC (%)	AUPR (%)
Tiresias (Fokoue *et al.* (2016)	—	—	—	91.5	93.9	—	—	68.0	—
E2E/GO (Gao *et al.* (2018)	78.3	—	—	80.8	92.2	—	75.3	89.4	—
E2E (Gao *et al.* (2018)	84.8	—	91.0	85.0	91.6	—	84.6	90.5	—
CPI-GNN (Tsubaki *et al.* (2019)	83.2	—	—	93.0	97.0	—	71.5	73.8	—
DrugVQA (Zheng *et al.* (2020)	88.7	93.6	—	91.0	96.0	—	86.0	92.2	—
GraphDTA (Nguyen *et al.* (2021)	85.5	93.6	93.4	94.7	98.2	97.8	74.4	82.7	82.7
TransformerCPI (Chen *et al.* (2020)	89.3	95.7	95.8	94.9	98.6	98.5	82.6	90.7	91.1
BridgeDPI	93.0	97.5	97.3	96.1	98.9	98.6	89.3	95.8	95.5

Θ is the set of all parameters in our model, λ is an adjustable regularization coefficient that balances the terms.

2.3 Implementation details

We use Pytorch 1.6.0 (Paszke *et al.*, 2019) to implement BridgeDPI. The Adam (Kingma and Ba, 2019) optimizer with learning rate 0.001 is used in our experiments. And the L2 regularization coefficient λ is set to 0.001. For each epoch, the data of protein–drug pairs are randomly shuffled and the batch size is set to 512. BridgeDPI will be trained for 100 epochs and the model with the best AUROC on the validation set will be retained. For the setting of other hyper-parameters, such as the number of layers, the number of neurons, the ratio of dropout and others, experiments are carried out to choose their values according to the performance on the validation set. All of the experimental processes are running on one NVIDIA GeForce RTX 1080 Ti GPU. And the final hyper-parameters' setting of BridgeDPI is shown in Table 1.

For proteins, we use a 1D CNN, which has 64 filters with width 25 to extract its local features. After that, a one-layer FFN is used to transform the local features to a 128-dimension space. For protein's global features, we introduce a two-layer FFN for non-linear transformation, containing 1024 and 128 neurons, respectively. For drugs, we employ a 1D CNN with 64 filters of width 25 to extract its local features. Then, a one-layer FFN is also used to transform the local features to the 128-dimension space. For drug's global features, we introduce a three-layer network for non-linear transformation, containing 1024, 256 and 128 neurons, respectively. Their outputs are all 128-dimension vectors that serve as the embeddings of protein/drug nodes on the bridge graph. For bridge graph, we introduce 64 bridge nodes with the same 128 dimensions as protein/drug nodes in the graph. Then, we use a three-layer GNN to capture the network-level information, which means that each node can aggregate at most three-depth neighbor information outwards. In the end, the scores of the interactions between proteins and drugs are obtained through a two-layer FFN with 128, 1 neurons. Moreover, we also use the dropout technique (Srivastava *et al.*, 2014) to improve BridgeDPI's generalization and the dropout rate is set to 0.5.

3 Results

3.1 Performance on the BindingDB dataset

To evaluate the performance of BridgeDPI in predicting DPIs, we compare BridgeDPI with some methods including Tiresias (Fokoue *et al.*, 2016), E2E (Gao *et al.*, 2018), CPI-GNN (Tsubaki *et al.*, 2019), DrugVQA (Zheng *et al.*, 2020), GraphDTA (Nguyen *et al.*, 2021) and TransformerCPI (Chen *et al.*, 2020) on the BindingDB dataset. In addition, we believe that a practical model should be able to handle unseen proteins because of a mass of unknown proteins in nature. Thus, we follow Zheng *et al.* (2020) to divide the test set into a seen protein test set and an unseen protein test set, and we investigate the performance of BridgeDPI in these two test sets.

Table 3. Performances of BridgeDPI and baselines on the C.ELEGANS and the HUMAN datasets

Methods	C.ELEGANS		HUMAN	
	AUROC (%)	F1 (%)	AUROC (%)	F1 (%)
k-NN	85.8	81.4	86.0	85.8
RF	90.2	83.2	94.0	87.9
LR	89.2	88.3	91.1	90.2
SVM	89.4	80.1	91.0	95.8
E2E/GO[a]	98.6	95.0	97.0	90.3
CPI-GNN	97.8	93.3	97.0	92.0
DrugVQA	—	—	97.9	95.7
GraphDTA	97.4	91.9	96.0	89.7
TransformerCPI	98.8	95.2	97.3	92.0
BridgeDPI	99.5	97.0	99.0	94.9

[a]GO feature required by E2E needs to be obtained from the UniProt database, and there are a large number of GO missing for these two datasets. Therefore, we only reproduced the results of the E2E/GO [without (the) GO feature] model.

Table 2 shows the results of BridgeDPI and other baselines on these two test sets. The symbol '—' indicates the absence of results, which means that the experimental results are not present in the cited paper. Overall, BridgeDPI achieves SOTA performance in the test set. The accuracy (ACC), AUROC and the area under the precision-recall curve (AUPR) of BridgeDPI have reached to 93.0%, 97.5% and 97.3%, respectively, which are significantly superior to the previous method.

For the seen protein test set, we find that the results in all models are decent, with AUROC and ACC generally exceeding 90%. Tiresias, which infers DPIs by constructing a large-scale association network, has a relatively low AUROC of 93.9%. The potential reason is that the DPIs do not necessarily exist between proteins and drugs with high similarity information. This also proves that it is not enough to rely solely on the 'guilt-by-association' principle to infer DPIs. Other learning-based approaches adequately learn the mechanism of DPIs by the priori DPI information, increasing the AUROC to around 97%, suggesting that learning to model the interaction mechanism is significant and necessary for DPI prediction (except for E2E, which we speculate the reason is that the E2E model is too partial to the performance on the unseen proteins but ignores the seen proteins). Among them, BridgeDPI achieves the best AUROC and ACC, reaching 98.9% and 96.1%, respectively. Compared to other baselines, BridgeDPI not only pays attention to learn the interaction mechanism but also combines a network-level perspective to assist learning. Consequently, our model can obtain more comprehensive feature expressions of proteins and drugs that aggregate the network-level information. And the expressions will be more conductive to the learning of the DPI mechanism.

For the unseen protein test set, it is clear that the difference in the performance of these methods varies greatly. Tiresias yields the worst AUROC of 68.0%, which means this method cannot be easily applied to unseen proteins. This is easy to understand because the unseen proteins are some isolated nodes in the drug–protein association network, lacking enough neighbor information to infer how they interact. Other learning-based approaches make some improvements by learning the interaction mechanism of drugs and proteins. Among them, BridgeDPI outperforms other methods and achieves SOTA performance, with AUROC and ACC reaching 95.8% and 89.3% in unseen proteins, 3.9% and 3.8% higher than the previous optimal method (DrugVQA). It indicates that introducing the bridge nodes indeed improves the performance on the unseen protein set by incorporating the network-level information and the learned interaction mechanism. Moreover, the constructed bridge graph also enables BridgeDPI to learn some deeper interaction rules because each information aggregation in the GNN is like an interaction among proteins or drugs, or between proteins and drugs. And this is why BridgeDPI is more accurate in predicting DPIs on both the seen protein test set and the unseen protein test set.

3.2 Performance on the C.ELEGANS and HUMAN datasets

Furthermore, we also conduct experiments on the C.ELEGANS dataset and HUMAN dataset (Liu *et al.*, 2015), which are widely used in many studies. We choose k-Nearest Neighbor (k-NN), Random Forest (RF), Logistic Regression (LR), Support Vector Machine (SVM), E2E/GO, CPI-GNN, DrugVQA, GraphDTA and TransformerCPI as the baselines. The results are shown in Table 3. Since Gao *et al.* (2018) do not provide the code of E2E, we reproduce their model and obtain the experimental results on the two datasets. We used the codes from GraphDTA's (Nguyen *et al.*, 2021) and TransformerCPI's (Chen *et al.*, 2020) github to generate their results on the BindingDB dataset. The results of other baselines are from their original papers. As can be seen from Table 3, for the randomly divided C.ELEGANS and HUMAN datasets, almost all proteins in the test set are seen proteins, which means that the models can learn almost all of the protein information well from the training dataset, resulting in very good results. In this case, the unsupervised k-NN is slightly worse than other models, with AUROC 85.8% and F1 81.4% on the C.ELEGANS dataset, AUROC 86.0% and F1 85.8% on the HUMAN dataset, respectively. In contrast, the traditional supervised machine learning methods (i.e. RF, LR and SVM) are slightly better, with AUROC of the C.ELEGANS dataset reaching around 90.0%, AUROC of the HUMAN dataset exceeding 91.0%. The deep learning methods E2E/GO, CPI-GNN, DrugVQA, GraphDTA, TransformerCPI and BridgeDPI all reach excellent performance, with AUROC over 96.0% and F1 over 89%. Among them, BridgeDPI achieves the best performance, with AUROC, F1 of 99.5%, 97.0% on the C.ELEGANS dataset, respectively, and 99.0%, 94.9% on the HUMAN dataset, respectively. The results are in line with our expectations. Because the models such as KNN, RF, LR and SVM, without high-quality features, are difficult to learn complex non-linear relationships among DPIs, while the deep learning models have strong feature extraction abilities to learn the interaction rules. On this basis, BridgeDPI integrates the network-level information and the learned interaction mechanism, further improving the results.

3.3 Performance on an independent test set

Although we have achieved excellent results on these benchmark datasets, such datasets have serious data bias, which will lead to the inflated performance (Chen *et al.*, 2019; Yang *et al.*, 2020). In order to investigate the realistic performance of our model, we conduct the following experiments: train models on the BindingDB dataset and test models on the DUD-E dataset. Additionally, we propose an evaluation indicators called pP@k, which is defined as the average protein-level precision at the top k predictions (as shown in Formula 12). pP@k indicates the accuracy of the model for k recalled drugs from the protein level, which can reasonably evaluate the reliability of DPI prediction methods in drug screening.

$$\mathrm{pP@}k = \frac{1}{m}\sum_{i=1}^{m}\frac{\sum_{j\in \hat{s}_r^k(i)} y_{(i-j)}}{k}, \tag{12}$$

where m is the number of proteins, $\hat{s}_r^k(i)$ is the set of k most possible drugs recalled by the model for protein i, $y_{(i-j)} \in \{0,1\}$ is the true interaction value for the pair of protein i and drug j.

We set $k = 10, 20, 40, 80, 160$ in this experiment. The results are shown in Table 4. Not surprisingly, the performances of these models are all greatly reduced, with AUROC of the SVM even less than 50%. Compared with other models, the performance of BridgeDPI is the best. For AUROC, BridgeDPI is 9.41%, 8.58%, 29.14%, 32.03%, 46.79% higher than E2E/GO, KNN, RF, LR, SVM, respectively. Moreover, if the whole BindingDB dataset is used for training, the AUROC of BridgeDPI and E2E/GO will reach to 77.2% and 74.8%, respectively. For pP@k, BridgeDPI can accurately recall more than 60% of drug candidates at $k \leq 80$, which significantly outperforms other compared methods. And if the whole BindingDB dataset is used for training, BridgeDPI can accurately recall more than 80% of drug candidates at $k \leq 40$. The results show the reliability of BridgeDPI and that BridgeDPI performs better even under a more realistic condition.

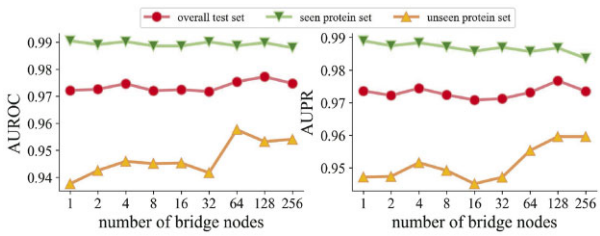

Fig. 3. Performance of BridgeDPI with different number of bridge nodes

Table 4. Performances of different methods on an independent test set

Methods	AUROC (%)	pP@10 (%)	pP@20 (%)	pP@40 (%)	pP@80 (%)	pP@160 (%)
Training on the customized BindingDB dataset (Gao *et al.*, 2018).						
k-NN	65.3	40.3	41.3	40.9	40.3	37.8
RF	54.9	42.0	41.2	40.5	40.0	37.6
LR	53.7	47.9	48.4	48.6	47.1	42.2
SVM	48.3	40.3	39.7	38.7	37.8	35.0
E2E/GO	64.8	53.1	54.8	54.8	52.7	46.4
BridgeDPI	70.9	68.2	67.5	66.1	61.6	53.1
Training on the full BindingDB dataset (Gilson *et al.*, 2016).						
E2E/GO	74.8	70.1	70.1	67.5	62.5	53.3
BridgeDPI	77.2	85.2	84.8	81.5	73.3	60.4

Table 5. Prediction results of possible antiviral drugs and viral targets

Molecules	Predicted probability (%)	Source
3CLpro		
Baricitinib	99.5	Kalil *et al.* (2021)
Remdesivir	97.0	Elfiky (2020)
Lopinavir	82.3	Stower (2020)
Ritonavir	55.5	Stower (2020)
Aspirin	3.7	
RdRp		
Ivermectin	99.6	Caly *et al.* (2020)
Remdesivir	98.3	Elfiky (2020)
Sofosbuvir	97.8	Sadeghi *et al.* (2020)
Daclatasvir	94.0	Sadeghi *et al.* (2020)
Lopinavir	87.5	Stower (2020)
Ritonavir	60.6	Stower (2020)
Aspirin	3.3	

3.4 Effect of the number of bridge nodes

By introducing the bridge nodes, BridgeDPI builds multiple bridges between all proteins and drugs. Thus, we carry out further study about the influence of the number of bridge nodes. We apply different numbers (i.e. 1, 2, 4, 8, 16, 32, 64, 128, 256) of bridge nodes to observe the performance on the BindingDB dataset. Figure 3 shows the results of the overall test set, the seen protein set and the unseen protein set for different numbers of bridge nodes. As we can see, the performances of different numbers of bridge nodes are stable on seen proteins but fluctuate greatly on unseen proteins. As the number of bridge nodes increases, the AUROC and AUPR of the unseen protein set are continuously improving, and thus it leads a better performance on the overall test set. This is in line with our respective because the bridge nodes' connection to all proteins and drugs means the unseen proteins are no longer an isolated node, which will make predicting DPIs easier. When the number of bridge nodes is 64, the performance is the best. More bridge nodes can explore more potential relationship between proteins and drugs together. However, too many bridge nodes will lead some nodes to play a similar role in voting and bring the risk of overfitting and excessive costs.

4 Case study

In order to show the performance of BridgeDPI in practical virtual screening, we select two important viral targets, 3C-like protease (3CLpro) and RNA-dependent RNA polymerase (RdRp), as the research objects. The two targets play a major role in the protein replication/transcription and host cell recognition, and therefore are vital for the viral reproduction and spread of infection (Murugan *et al.*, 2020). As same as (Kim *et al.*, 2021), we also select some candidates, such as Baricitinib and Ivermectin, to test the predicted interactions. First, we obtain the amino acid sequences of the two targets [the sequence of 3CLpro is from the Protein Data Bank (PDB) database (Sussman *et al.*, 1998) with PDB ID 6WQF, RdRp is from the National Center for Biotechnology Information (NCBI; Pruitt *et al.*, 2007) with NCBI YP_009725307.1]. Then, the sequences and the candidate molecules are fed into BridgeDPI. Finally, the interaction probabilities of them are predicted, as shown in Table 5.

Table 5 shows that Baricitinib, Remdesivir, Lopinavir and Ritonavir are all very potential drugs that can interact with the 3CLpro; Ivermectin, Redmdesivir, Sofosbuvir, Daclatasvir, Lopinavir and Ritonavir are all effective drugs that can bind with the RdRp. In fact, many studies and clinical trials also validate the results (Caly *et al.*, 2020; Elfiky, 2020; Favalli *et al.*, 2020; Kalil *et al.*, 2021; Sadeghi *et al.*, 2020; Stower, 2020). In contrast, unrelated drugs such as Aspirin have little interaction potential with the viral targets. These experimental results verify the validity and reliability of BridgeDPI in predicting new drugs, indicating that BridgeDPI plays a guide role in actual research and drug discovery.

5 Conclusion

In this study, we propose an end-to-end deep learning framework to predict DPIs by introducing the network-level information to a learning-based framework. We construct a supervised drug–protein network and introduce a class of bridge nodes to it. The bridge nodes bridge the gap among drugs and proteins by information passing among diverse drugs and proteins, and thus we can use a GNN to capture the network-level information and rely on a supervised 'guilt-by-association' to perform predictions. Therefore, our model integrates more comprehensive features by taking into account both the advantages of network-based methods and learning-based methods. The experiments show that our approach outperforms other competing methods on BindingDB, C.ELEGANS, Human, DUD-E datasets and achieves SOTA performances. Moreover, the case study with concrete examples also reaffirms the usefulness of our model.

Funding

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61832019, and the Human Provincial Science and Technology Program [2019CB1007 and 2021RC4008].

Conflict of Interest: none declared.

References

Avorn,J. (2015) The $2.6 billion pill–methodologic and policy considerations. *N. Engl. J. Med.*, **372**, 1877–1879.

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

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

Caly,L. *et al.* (2020) The FDA-approved drug ivermectin inhibits the replication of SARS-CoV-2 in vitro. *Antiviral Res.*, **178**, 104787.

Chen,L. *et al.* (2019) Hidden bias in the dud-e dataset leads to misleading performance of deep learning in structure-based virtual screening. *PLoS One*, **14**, e0220113.

Chen,L. *et al.* (2020) TransformerCPI: improving compound–protein interaction prediction by sequence-based deep learning with self-attention mechanism and label reversal experiments. *Bioinformatics*, **36**, 4406–4414. [CrossRef][*10.1093/bioinformatics/btaa524*]

Ding,H. *et al.* (2014) Similarity-based machine learning methods for predicting drug–target interactions: a brief review. *Brief. Bioinform.*, **15**, 734–747.

Durrant,J.D. and McCammon,J.A. (2011) NNScore 2.0: a neural-network receptor–ligand scoring function. *J. Chem. Inf. Model.*, **51**, 2897–2903.

Elfiky,A.A. (2020) Ribavirin, remdesivir, sofosbuvir, galidesivir, and tenofovir against SARS-CoV-2 RNA dependent RNA polymerase (RdRp): a molecular docking study. *Life Sci.*, **253**, 117592.

Favalli,E.G. *et al.* (2020) Baricitinib for COVID-19: a suitable treatment? *Lancet Infect. Dis.*, **20**, 1012–1013.

Fokoue,A. *et al.* (2016). Predicting drug-drug interactions through large-scale similarity-based link prediction. In: *European Semantic Web Conference*, Crete, Greece, Springer, pp. 774–789.

Gao,K.Y. *et al.* (2018) Interpretable drug target prediction using deep neural representation. In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)*, International Joint Conferences on Artificial Intelligence, Stockholm, pp. 3371–3377.

Gilson,M.K. *et al.* (2016) Bindingdb in 2015: a public database for medicinal chemistry, computational chemistry and systems pharmacology. *Nucleic Acids Res.*, **44**, D1045–D1053.

Gschwend,D.A. *et al.* (1996) Molecular docking towards drug discovery. *J. Mol. Recogn.*, **9**, 175–186.

Kalil,A.C. *et al.*; ACTT-2 Study Group Members. (2021) Baricitinib plus remdesivir for hospitalized adults with covid-19. *N. Engl. J. Med.*, **384**, 795–807.

Kim,Q. *et al.* (2021) Bayesian neural network with pretrained protein embedding enhances prediction accuracy of drug-protein interaction. *Bioinformatics*, **37**, 3428–3435.
Kingma,D.P. and Ba,J.A. (2019). A method for stochastic optimization. arxiv 2014. *arXiv preprint arXiv:1412.6980*, 434. https://doi.org/10.48550/arXiv.1412.6980.
Led,P. and Caflisch,A. (2018) Protein structure-based drug design: from docking to molecular dynamics. *Curr. Opin. Struct. Biol.*, **48**, 93–102.
Leslie,C.S. *et al.* (2004) Mismatch string kernels for discriminative protein classification. *Bioinformatics*, **20**, 467–476.
Li,M. *et al.* (2022) BACPI: a bi-directional attention neural network for compound-protein interaction and binding affinity prediction. *Bioinformatics*, **38**, 1995–2002.
Li,S. *et al.* (2020) MONN: a multi-objective neural network for predicting compound-protein interactions and affinities. *Cell Syst.*, **10**, 308–322.
Liu,H. *et al.* (2015) Improving compound–protein interaction prediction by building up highly credible negative samples. *Bioinformatics*, **31**, i221–i229.
Liu,T. and Altman,R.B. (2015) Relating essential proteins to drug side-effects using canonical component analysis: a structure-based approach. *J. Chem. Inf. Model.*, **55**, 1483–1494.
Luo,H. *et al.* (2019) A novel drug repositioning approach based on collaborative metric learning. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, 18, 463–471.
Luo,H. *et al.* (2021) Biomedical data and computational models for drug repositioning: a comprehensive review. *Brief. Bioinform.*, **22**, 1604–1619.
Luo,Y. *et al.* (2017) A network integration approach for drug-target interaction prediction and computational drug repositioning from heterogeneous information. *Nat. Commun.*, **8**, 1–13.
Maggiora,G.M. *et al.* (2014) Molecular similarity in medicinal chemistry. *J. Med. Chem.*, **57**, 3186–3204.
Mizianty,M.J. *et al.* (2014) Covering complete proteomes with X-ray structures: a current snapshot. *Acta Crystallogr. D Biol. Crystallogr.*, **70**, 2781–2793.
Murugan,N.A. *et al.* (2020) Searching for target-specific and multi-targeting organics for Covid-19 in the Drugbank database with a double scoring approach. *Sci. Rep.*, **10**, 1–16.
Mysinger,M.M. *et al.* (2012) Directory of useful decoys, enhanced (dud-e): better ligands and decoys for better benchmarking. *J. Med. Chem.*, **55**, 6582–6594.
Nguyen,T. *et al.* (2021) GraphDTA: predicting drug–target binding affinity with graph neural networks. *Bioinformatics*, **37**, 1140–1147.
Paszke,A. *et al.* (2019) PyTorch: an imperative style, high-performance deep learning library. In: *Advances in Neural Information Processing Systems*, pp. 8026–8037.
Paul,S.M. *et al.* (2010) How to improve R&D productivity: the pharmaceutical industry's grand challenge. *Nat. Rev. Drug Discov.*, **9**, 203–214.
Pruitt,K.D. *et al.* (2007) NCBI reference sequences (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Res.*, **35**, D61–D65.
Ramsundar,B. *et al.* (2019). *Deep Learning for the Life Sciences*. O'Reilly Media, Inc., Sebastopol.
Rogers,D. and Hahn,M. (2010) Extended-connectivity fingerprints. *J. Chem. Inf. Model.*, **50**, 742–754.
Sadeghi,A. *et al.* (2020) Sofosbuvir and daclatasvir compared with standard of care in the treatment of patients admitted to hospital with moderate or severe coronavirus infection (Covid-19): a randomized controlled trial. *J. Antimicrob. Chemother.*, **75**, 3379–3385.
Srivastava,N. *et al.* (2014) Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, **15**, 1929–1958.
Stower,H. (2020) Lopinavir–ritonavir in severe Covid-19. *Nat. Med.*, **26**, 465.
Sussman,J.L. *et al.* (1998) Protein Data Bank (PDB): database of three-dimensional structural information of biological macromolecules. *Acta Crystallogr. D Biol. Crystallogr.*, **54**, 1078–1084.
Tsubaki,M. *et al.* (2019) Compound–protein interaction prediction with end-to-end learning of neural networks for graphs and sequences. *Bioinformatics*, **35**, 309–318.
Wan,F. *et al.* (2019) NeoDTI: neural integration of neighbor information from a heterogeneous network for discovering new drug–target interactions. *Bioinformatics*, **35**, 104–111.
Wang,C. and Lukasz,K. (2019) Review and comparative assessment of similarity-based methods for prediction of drug-protein interactions in the druggable human proteome. *Brief. Bioinform.*, **20**, 2066–2087. [CrossRef][*10.1093/bib/bby069*]
Wang,K. *et al.* (2021) DeepDTAF: a deep learning method to predict protein–-ligand binding affinity. *Brief. Bioinform.* **22**, bbab072.
Wang,Y. and Zeng,J. (2013) Predicting drug-target interactions using restricted Boltzmann machines. *Bioinformatics*, **29**, i126–i134.
Weininger,D. (1988) Smiles, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.*, **28**, 31–36.
Yang,J. *et al.* (2020) Predicting or pretending: artificial intelligence for protein-ligand interactions lack of sufficiently large and unbiased datasets. *Front. Pharmacol.*, **11**, 69.
Yuvaraj,N. *et al.* (2021) Analysis of protein-ligand interactions of SARS-CoV-2 against selective drug using deep neural networks. *Big Data Min. Anal.*, **4**, 76–83.
Zhang,Q.C. *et al.* (2012) Structure-based prediction of protein–protein interactions on a genome-wide scale. *Nature*, **490**, 556–560.
Zheng,S. *et al.* (2020) Predicting drug–protein interaction using quasi-visual question answering system. *Nat. Mach. Intell.*, **2**, 134–140.