BridgeDPI: A Novel Graph Neural Network for Predicting Drug-Protein Interactions

Yifan Wu\textsuperscript{1, 2}, Min Gao\textsuperscript{1}, Min Zeng\textsuperscript{2}, Feiyang Chen\textsuperscript{1}, Min Li\textsuperscript{2,*} and Jie Zhang\textsuperscript{1, 3,*}

\textsuperscript{1}SenseTime Research, Shanghai, 200233, China, \textsuperscript{2}School of Computer Science and Engineering, Central South University, Changsha, 410083, China, \textsuperscript{3}Qingyuan Research Institute, Shanghai Jiao Tong University, Shanghai, China

\textsuperscript{†}These authors contributed equally to this work.

\textsuperscript{*}To whom correspondence should be addressed.

Abstract

Motivation: Exploring drug-protein interactions (DPIs) work as a pivotal step in drug discovery. The fast expansion of available biological data enables computational methods effectively assist in experimental methods. Among them, deep learning methods extract features only from basic characteristics, such as protein sequences, molecule structures. Others achieve significant improvement by learning from not only sequences/molecules but the protein-protein and drug-drug associations (PPAs and DDAs). The PPAs and DDAs are generally obtained by using computational methods. However, existing computational methods have some limitations, resulting in low-quality PPAs and DDAs that hamper the prediction performance. Therefore, we hope to develop a novel supervised learning method to learn the PPAs and DDAs effectively and thereby improve the prediction performance of the specific task of DPI.

Results: In this research, we propose a novel deep learning framework, namely BridgeDPI. BridgeDPI introduces a class of nodes named hyper-nodes, which bridge different proteins/drugs to work as PPAs and DDAs. The hyper-nodes can be supervised learned for the specific task of DPI since the whole process is an end-to-end learning. Consequently, such a model would improve prediction performance of DPI. In three real-world datasets, we further demonstrate that BridgeDPI outperforms state-of-the-art methods, achieving AUC of 0.989 (for seen proteins), 0.952 (for unseen proteins) in the customized BindingDB dataset, 0.995 in the C.elegans dataset, 0.990 in the human dataset. Moreover, ablation studies verify the effectiveness of the hyper-nodes. Last, in an independent verification, BridgeDPI explores the candidate bindings among COVID-19’s proteins and various antiviral drugs. And the predictive results accord with the statement of the World Health Organization and Food and Drug Administration, showing the validity and reliability of BridgeDPI. Hopefully, the BridgeDPI method will facilitate and accelerate real-world drug discovery and drug screening.

Availability: The source code of BridgeDPI can be accessed at https://github.com/DeepAAI/BridgeDPI.

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

1 Introduction

The drug discovery and drug screening are complex. The typical timeline usually takes 10-20 years and costs US$0.5-2.6 billion (Avorn et al., 2015; Paul et al., 2010). Among them, exploring possible drug-protein interactions (DPIs) is a crucial step. Although experimental assays remain the most reliable approach for determining DPIS, they are time-consuming and cost-intensive. Therefore, efficient computational methods for predicting protein-drug interactions are significant and urgently demanded.

Current DPI prediction methods can be summarized as three forms: docking-based methods, machine learning-based methods, and deep learning-based methods. Docking-based methods look for the best binding
Deep learning models such as DeepDTA (Öztürk et al., 2008) and DPI prediction by using the deep structure and vast learnable parameters. However, one needs to choose, combine and compare these handcrafted features carefully, which require expertise and experience. These methods usually use handcrafted features before modeling, which require much expertise. Recently, with the plentiful accumulation of data, deep learning-based methods have been successfully applied to various bioinformatics tasks (Zeng et al., 2019; A et al., 2020; Min et al., 2017; Zhang et al., 2019, 2020). It has further improved the performance of DPI prediction by using the deep structure and vast learnable parameters. Deep learning models such as DeepDTA (Öztürk et al., 2018), WideDTA (Öztürk et al., 2019), DeepConv-DTI (Lee et al., 2019), PADME (Feng et al., 2018), GraphDTA (Nguyen et al., 2019), E2E (Gao et al., 2018), drugVQA (Zeng et al., 2020) have similar steps: 1) encode proteins and drugs. 2) design a feature extractor modules to capture high-level features of proteins and drugs. 3) fuse the high-level features of proteins and drugs, and perform prediction through full connected layers. The disadvantage of these methods is that they neglect to leverage the protein-protein associations (PPAs) and drug-drug associations (DDAs). The magic of these methods is that they neglect to leverage the protein-protein associations (PPAs) and drug-drug associations (DDAs). For the methods based on known DTIs, it uses negative sampling to generate negative samples for PPAs and DDAs without any reliable biological evidence. However, negative sampling treat the unknown drug-protein pairs as negative. However, the negative drug-protein pairs could be positive.

To tackle the above limitations and further improve the prediction performance, we develop BridgeDPI, a deep learning framework for predicting DPs. The novelty of BridgeDPI is that we introduce the hyper-nodes to connect different proteins/drugs. The role of hyper-nodes is to construct bridges between proteins/drugs. The bridges implicitly measure the associations among proteins/drugs and therefore can be used as the networks of PPAs and DDAs. On one side, the hyper-nodes are automatically learned for DPI prediction. The quality of the learned PPAs and DDAs is guaranteed by the back propagation of the end-to-end learning of BridgeDPI. On the other side, some unknown biological connections may also be explored by the hyper-nodes. In brief, the hyper-nodes are considered as the essential elements to improve the prediction performance.

We demonstrate the superior performance of BridgeDPI the AUC scores of 0.989 (for seen proteins), 0.952 (for unseen proteins) in the customized BindingDB dataset, 0.995 in the C.elegans dataset, 0.990 in the human dataset.

2 Methods
2.1 Overview
The overview of BridgeDPI is shown in Figure 1. BridgeDPI contains two inputs: the protein sequence and the drug smiles. For protein sequences, a three-layer feed-forward network is used to process the k-mer features. In addition, a Convolutional Neural Network (CNN) is used to extract the sequence features of drugs. We combine the two features to obtain the final protein embeddings. For drug smiles, a two-layer feed-forward network is used to process the molecular fingerprint features. In addition, a CNN is used to extract the sequence features of drugs. We also combine the two features to obtain the final drug embeddings. After that, we introduce some hyper-nodes to construct the bridges between proteins/drugs and use Graph Neural Network (GNN) to learn the associations between proteins/drugs. Finally, we get element-wise products of the protein and drug outputs after GNN, and then use a linear layer with sigmoid activation to predict the interactions.

![Diagram of BridgeDPI model](image)
2.2 Embedding of proteins and drugs

Before feeding protein sequences and drug molecules to BridgeDTI, they need to be encoded as numeric vectors. For a long time in the past, the k-mer and FingerPrint (FP) feature have been very effective to represent proteins and drugs. Since the advent of deep learning techniques, people begin to characterize proteins and drugs at the amino acid level and atomic level by using CNN, GNN, etc. However, in fact, k-mer and FP features cannot be replaced completely by the sequence or graph features. Thus, we reintroduce k-mer and FP features, which are effective but ignored by many researchers at present.

For proteins, k-mer is a classical method for protein embedding. K-mer features can describe the type and number of amino acid functional groups, which are very important in the prediction of DPs. As a result, k-mer features are used to represent the proteins. In our study, we set k as 1, 2, 3, and thus we got \(20^1 + 20^2 + 20^3 = 8420\) k-mer features.

\[
x^d_i = (x_{i1}^d, x_{i2}^d, \ldots, x_{in}^d)
\]

where \(x_{id} = \text{represents } 1\text{-mer feature of protein } i, \text{respectively. We normalize each part horizontally to eliminate the effect of protein length, as shown in Formula (2).}

\[
\text{normalized}(x_{id}) = \frac{x_{id} - \text{mean}(x_{id})}{\text{std}(x_{id})}
\]

For drug molecules, FP is a very efficient technique in drug discovery and virtual screening. In molecular FP, the topological information of molecular structure is encoded as a vector, which can well represent a drug molecule. Consequently, we use Morgan FP (Cereto-Massagué et al., 2015) to represent a molecule. It can encode a drug as a vector with dimensions of 1024.

\[
F^j = (b_1, b_2, \ldots, b_{1024})
\]

where \(F^j\) represents FP features of drug \(j, b_1, b_{1024}\) are the binary values in the fingerprint.

After getting the characterization of proteins \(A^i\) and drugs \(F^j\), their relationship cannot be constructed directly. Because the two vectors have different dimensions and do not belong to the same space. Therefore, they need to be mapped into the same space through several layers of neural networks. At the same time, we use CNN with max-pooling to extract the sequence features of proteins and drugs, respectively, and then fuse them together.

\[
u^i = L_{p}(A^i) + p^i
\]

\[
v^j = L_{d}(F^j) + d^j
\]

where \(L_{p}()\) and \(L_{d}()\) are the nonlinear transformation layers for protein and drug, respectively, \(p^i\) and \(d^j\) are the sequence features extracted by CNNs. Finally, \(u^i\) is used as the embedding of protein \(i, v^j\) is used as the embedding of drug \(j\).

2.3 Introduction of hyper-nodes

After the embedding of proteins and drugs, how can we predict the interactions between proteins and drugs, especially those unconnected protein-drug pairs candidates? It is the main challenge to DPP predictions due to the lack of sufficient neighborhood information. As shown in Figure 2, given known interacted pairs, we need to infer unknown relationship between protein \(p_2\) and drug \(d_2\). Traditional methods usually computed DDAs and PPAs to construct a heterogeneous network. And then according to the associations between protein/drugs, we can construct a path \(p_2 - p_1 - d_1 - d_2\) to infer the relationship. However, as mentioned above, it is difficult to acquire high-quality PPAs and DDAs. Therefore, we introduce a new kind of virtual nodes, namely hyper-nodes. These nodes connect all proteins and drugs, which are randomly initialized and updated during the training process.

With these hyper-nodes, we can construct some bridges between proteins/drugs and capture the relationship between them. It makes our model learn a higher-quality PPA and DDA information from the data and further improve the predictive performance of DTI. As illustrated in Figure 2, although we do not know the relationship between \(p_1\) and \(p_2\), by introducing a hyper-node \(h_1\), we can get the relationship between \(p_1\) and \(h_1\), \(p_2\) and \(h_1\). Therefore, the relationship between \(p_1\) and \(p_2\) can be derived from \(p_2 - h_1 - p_1\). In the same way, the relationship between \(d_1\) and \(d_2\) can be derived from \(d_2 - h_2 - d_1\). Finally, the relationship between \(p_2\) and \(d_2\) can be inferred through \(p_2 - h_1 - p_1 - d_1 - h_2 - d_2\).

2.4 Graph neural network

Assume that the set of hyper-nodes is \(\{n_1, n_2, \ldots, n_m\}\), where \(m\) is the number of hyper-nodes, and hyper-nodes have the same embedding size with \(u^i\) and \(v^j\). For each protein-drug pair in dataset, we can get a graph including nodes \(u^i, v^j, n_1, n_2, \ldots, n_m\) with edges calculated by cosine similarity.

\[
A_{ij} = \frac{u_i \cdot n_j}{||u_i|| ||n_j||}
\]

where \(n_i\) and \(n_j\) represent any two nodes in the graph (including \(u^i\) and \(v^j\)), \(||\cdot||\) is to compute 2-norm of a vector. In fact, \(A_{ij}\) could be negative, which are not suitable for the calculation of GNN. Thus, we need to filter out the negative edges by \(ReLU(\cdot)\), which means there’s no edge if cosine similarity is less than zero. And the computational steps of GNN are as follows:

\[
L = D^{-1/2}ReLU(A)D^{-1/2}
\]

\[
Z_0 = (u^i, v^j, n_1, n_2, \ldots, n_m)^T
\]

\[
Z_t = \text{ReLU}(LZ_{t-1}(W^t + b^t)) + Z_{t-1}^\perp
\]

where \(A \in \mathbb{R}^{(m+2) \times (m+2)}\) is the similarity matrix, \(Z_0 \in \mathbb{R}^{(m+2) \times d}\) is the embedding matrix of nodes, \(d\) is the embedding size of
nodes. \( D \in \mathbb{R}^{(m+2) \times (m+2)} \) is the degree matrix. \( Z^t \in \mathbb{R}^{(m+2) \times d} \) is the output of \( t \)-th layer of GNN, \( W^t \) and \( b^t \) are the parameters in \( t \)-th layer of GNN. Additionally, residual structure (He et al., 2016) is applied in each layer of GNN, as shown in Equation (9). It is a very effective structure in deep networks because it guarantees that some shallower network may exist in the model.

2.5 Prediction of protein-drug interactions

The output of GNN belongs to \( \mathbb{R}^{(m+2) \times d} \), and the first two rows are the embeddings of protein nodes and drug nodes after fusing graph information. The two embeddings are taken out to do element-wise multiplication and then fed to a fully connected layer. Finally, the sigmoid() is applied to classify whether the protein-drug pairs has interactions.

\[
\begin{align*}
    h_{ij} &= \hat{u}^i \odot \hat{v}^j \\
    \tilde{y}_{ij} &= \text{sigmoid}(f_s(h_{ij}))
\end{align*}
\]

where \( \hat{u}^i \) and \( \hat{v}^j \) are the first two rows of the final output matrix in GNN, \( f_s(\cdot) \) is the fully connected layer, \( g_{ij} \) represents the probability of the interaction between protein \( i \) and drug \( j \), \( \odot \) means element-wise product.

2.6 Other details

Batch normalization (Ioffe and Szegedy, 2015), which standardizes each batch of data, is a very effective method to accelerate training and improve the performance of the deep network. Dropout (Srivastava et al., 2014), which randomly drops out of neuron units, can greatly improve the generalization of the model. We add batch normalization and dropout after each layer in the model. Finally, the model is trained by minimizing the cross-entropy loss function.

\[
L(y_{ij}, \tilde{y}_{ij} | \theta) = -(y_{ij} \log \tilde{y}_{ij} + (1 - y_{ij}) \log(1-\tilde{y}_{ij}) + \lambda \sum_{\theta \in U} |\theta|^2
\]

where \( \theta \) is the set of all model parameters, \( \lambda \) is the regularization coefficient.

3 Experiments

3.1 Datasets

**BindingDB dataset.** BindingDB is a public, web-accessible database for measuring binding affinities, focusing on interactions between proteins and small drug molecules. It contains a total of 2,067,981 binding data, containing 1024, 256, 128 neurons, respectively. The outputs are 128-dimensional vectors that serve as the input of node embeddings on the complete DGI graph. Then, we use a 3-layers GNN, which means that each node has aggregated three layers of neighbor information. In the end, the scores of the interactions between proteins and drugs are obtained through a 2-layers feed forward network. Moreover, we dropout 50% neurons after each layer. And the L2 regularization coefficient is set to 0.001 to limit the model ability and prevent over-fitting.

For proteins, we introduce a 2-layers network for nonlinear transformation, containing 1024, 128 neurons, respectively. For drug molecules, we introduce a 3-layers network for nonlinear transformation, containing 1024, 256, 128 neurons, respectively. The outputs are 128-dimensional vectors that serve as the input of node embeddings on the complete DI graph. Then, we use a 3-layers GNN, which means that each node has aggregated three layers of neighbor information. In the end, the scores of the interactions between proteins and drugs are obtained through a 2-layers feed forward network. Moreover, we dropout 50% neurons after each layer. And the L2 regularization coefficient is set to 0.001 to limit the model ability and prevent over-fitting.

3.2 Training details

The proposed model is implemented with Pytorch 1.6.0 (Paszke et al., 2019) and the parameters are initialized by default. We use Adam (Kingma and Ba, 2019) optimizer with a learning rate of 0.001 to adjust the parameters in the training process. In order to prevent over-fitting, L2 regularization is added to the loss function. For each step, a batch of protein-drug pairs is randomly selected to run the gradient descent algorithm and the batch size is set to 512. For the setting of other hyper-parameters, such as the number of layers, the number of neurons, the ratio of dropout, etc., many experiments are carried out to choose the values according to the performances on the validation set. The maximum number of epochs is set to 100, and the models with the best area under ROC curve AUC in the validation set are saved. Finally, the saved models are evaluated on a test set by metrics of accuracy (ACC) and the AUC. Besides, our model has low computational complexity, high parallelism, and fast training speed. The training process of the customized BindingDB dataset will be finished in about 15 minutes on the GPU platform of GTX 1080ti.

3.3 Results

First of all, we conduct experiments on the customized bindingDB dataset which is extracted from the bindingDB by Gao et al. Following their settings, we also make the same division of the dataset to ensure that the validation set and test set contain some unseen proteins that are not appear in the training set), which is closer to the real world. Based on the prediction results in the test set, we calculate AUC and ACC (threshold is 0.5, same as below) of BridgeDPI with the best parameters. To compare the performance with other methods, we choose Tiensias (Fokoue et al., 2016), DBN (Wen et al., 2017), GNN (Tsubaki et al., 2019), E2E (Gao et al., 2018) and DrugVQA (Zheng et al., 2020) as baselines. As we know, there are a large number of unknown proteins in nature, and this is why we should focus on predicting the new proteins, i.e., cold-start problem. Therefore, the test set is divided into an unseen protein set (the proteins that do not appear in the training set) and a seen protein set (the proteins that appear in the training set). As shown in Figure 3, the existing
models generally achieve good performances in the seen protein set (AUC exceeds 0.9, ACC exceeds 0.85), but differ greatly in the unseen protein set. Based on a traditional relational network, Tiresias performs poorly on the unseen proteins, with an AUC of only 0.68. Tiresias is a method based on computed similarity information to obtain the features of proteins and drugs, and it predicts DPIs through a logistic regression model. We hypothesize that only based on a linear model using similarity information, the expression of the unseen proteins may not be sufficient, resulting in the ACC on unseen proteins is even less than 0.5. DBN, E2E, DrugVQA, and BridgeDPI have AUC over 0.9 and ACC over 0.8 in unseen proteins, indicating the effectiveness of deep learning techniques in solving unseen proteins. Among them, our BridgeDPI outperforms other baselines and achieves state-of-the-art performances, with AUC and ACC reaching 0.987 and 0.954 in seen proteins, 0.951 and 0.887 in unseen proteins. It indicates that the introduction of hyper-nodes indeed improves the expression of protein/drug features, and the deep graph neural network also enables BridgeDPI to learn the deeper interaction rules between proteins and drugs.

Furthermore, we also conduct experiments on the C.elegans dataset and human dataset (Liu et al., 2015) which are widely used in many studies. The results are shown in Table 1. Since Gao et al. (Gao et al., 2018) do not provide the code of E2E, we reproduced the model and obtain experimental results on the two datasets without using the Gene Ontology (GO) features. Other results of baselines are from their original papers (Tsubaki et al., 2019; Zheng et al., 2020). From Table 1, for randomly divided C.elegans and human datasets, almost all proteins in the test set are seen, which means models can learn all protein information better from the training dataset, resulting in very good results. In this case, the unsupervised k-NN is slightly worse than other models, with AUC 0.858 and F1 0.814 on the C.elegans dataset. AUC 0.860 and F1 0.858 on the human dataset, respectively. The supervised machine learning methods (RF, L2, SVM) are slightly better, with AUC of the C.elegans dataset reaching around 0.9, AUC of the human dataset exceeding 0.9. In contrast, GNN, E2E/GO, DrugVQA, and BridgeDPI based on deep learning methods perform very well, with AUCs over 0.97 and F1s over 0.9. Among them, BridgeDPI achieves the best performances, with AUC, precision, recall, and F1 of 0.995, 0.980, 0.965, 0.972 on the C.elegans dataset, respectively, and 0.990, 0.963, 0.949, 0.956 on the human dataset, respectively. The results are in line with our expectations. Because the models such as KNN, RF, L2, and SVM, without high-quality features, are difficult to learn complex nonlinear relationships (protein-drug interaction), while the deep learning models have strong feature extraction abilities to learn the interaction rules. On this basis, BridgeDPI integrates PPA and DDA information, further improving the results.

Table 1. Performances of BridgeDPI with other models on different datasets

| Models       | AUC C.elegans | Precision C.elegans | Recall C.elegans | F1 C.elegans |
|--------------|---------------|---------------------|-----------------|--------------|
| k-NN         | 0.858         | 0.801               | 0.827           | 0.814        |
| RF           | 0.902         | 0.821               | 0.844           | 0.832        |
| L2           | 0.892         | 0.890               | 0.877           | 0.881        |
| SVM          | 0.894         | 0.785               | 0.818           | 0.801        |
| GNN          | 0.978         | 0.938               | 0.929           | 0.933        |
| E2E/GO       | 0.986         | 0.950               | 0.950           | 0.950        |
| BridgeDPI    | 0.995         | 0.980               | 0.965           | 0.972        |

| Models       | AUC Human     | Precision Human    | Recall Human    | F1 Human     |
|--------------|---------------|-------------------|----------------|-------------|
| k-NN         | 0.860         | 0.798             | 0.927           | 0.858       |
| RF           | 0.940         | 0.861             | 0.897           | 0.879       |
| L2           | 0.911         | 0.891             | 0.913           | 0.902       |
| SVM          | 0.910         | 0.966             | 0.950           | 0.958       |
| GNN          | 0.970         | 0.923             | 0.918           | 0.920       |
| E2E/GO       | 0.970         | 0.893             | 0.914           | 0.903       |
| DrugVQA      | 0.979         | 0.954             | 0.961           | 0.957       |
| BridgeDPI    | 0.990         | 0.963             | 0.949           | 0.956       |

Although we have achieved excellent results on these benchmark datasets, such datasets have serious data bias, which will lead to the inflated performances (Chen et al., 2019; Yang et al., 2020). In order to verify the realistic performances of our model, we conduct the following experiments: train models on the customized BindingDB dataset and test models on the DUD-E dataset. We conduct 5-fold cross-validation experiments: train models on the customized BindingDB dataset and verify the realistic performances of our model, we conduct the following experiments: train models on the customized BindingDB dataset and test models on the DUD-E dataset. We conduct 5-fold cross-validation experiments.
experiments on the customized BindingDB dataset to obtain 5 models, and then predict on the DUD-E dataset to obtain 5 results, last the 5 results are averaged to evaluate the performances. The results are shown in Figure 4. Similar to the previous reasons, the results for DrugVQA and E2E are not obtained. Not surprisingly, the performances of these models are greatly reduced, with AUC of the SVM even less than 0.5. Compared with other models, the AUC (0.709) of BridgeDPI is the best, which is 9.41%, 8.58%, 29.14%, 32.03%, 46.79% higher than E2E/GO, KNN, RF, L2, SVM, respectively. Moreover, if the whole BindingDB dataset is used for training, the AUC of BridgeDPI and E2E/GO will reach to 0.772 and 0.748. The results show the effectiveness of hyper-nodes and that BridgeDPI performs better even under more realistic conditions.

3.4 Ablative analysis

By introducing the hyper-nodes, BridgeDPI builds many bridges between all proteins/drugs to conduct the prediction of DPI. In order to discover the role of the hyper-nodes, we carry out a further ablation study on the customized BindingDB dataset about the influence of the number of hyper-nodes in our model. We set the number of hyper-nodes to -1, 1, 2, 4, 8, 16, 32, 64, 128, 256 separately to observe the results of our model on the test set. Among them, -1 means to remove the hyper-nodes from BridgeDPI, and the extracted feature vectors of proteins and drugs are multiplied directly and fed to the final fully connect layer. The overall AUC/ACC and the unseen proteins’ AUC/ACC are mainly focused, as shown in Figure 5. We can see that the introduction of the hyper-nodes can indeed improve the prediction performance of DPI, either in the overall AUC and ACC or in the unseen proteins’ AUC and ACC. As the number of hyper-nodes increases, the performances are further improved. When the number is at about 64, the overall AUC and unseen proteins’ AUC have reached the best values, 0.973 and 0.951 respectively. We speculate that more hyper-nodes mean more bridges between proteins/drugs. The more hyper-nodes measure the relationship between proteins/drugs together and play a similar role of voting. However, too many hyper-nodes will bring excessive costs and the risk of over-fitting. Therefore, we end up introducing 64 hyper-nodes into BridgeDPI.

3.5 Case study

The outbreak of COVID-19 has caused untold damage to human society, and scientists are working hard on drug discovery. The gene sequence of COVID-19 has already been detected. In order to verify the effectiveness of BridgeDPI in practical problems, we test a variety of interactions between current possible antiviral drugs and the protein targets translated from COVID-19’s viral gene fragments.
proteins/drugs so that the information of PPA and DDA can be captured, and then get the prediction of DTIs. The experiments show that our approach outperforms other competing methods on the customized BindingDB, C.elegans, Human, DUD-E datasets and achieves state-of-the-art performances. In order to verify the realistic performances of our model, we perform cross-validation experiments on the different datasets (training on BindingDB dataset, testing on DUDE dataset) and achieve a great result. Finally, the case study with concrete examples reenforces the usefulness of our model.

Acknowledgements

Funding

This work is supported in part by the NSFC-Zhejiang Joint Fund for the Integration of Industrialization and Informatication under Grant No. U1909208, Human Provincial Science and Technology Program 2015C61007.

References

A. Y. W., A. M. Z., A. Z. F., A. Y. Y., B. F. X. W., and A. M. L. (2020). Kaic: A knowledge attention-based deep learning framework for automatic icd coding. Neurocomputing. Avorn, J. et al. (2015). The Z2.6 billion pill-medicological policy and considerations.

N Engl J Med. 372(20), 1877–1879.

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

Bandini, A., Souiai, O., and Brun, C. (2017). Network analysis and protein function prediction with the prediction web site. Methods in Molecular Biology, 180, 313.

Beigel, J., Chung, K. K., Colombo, R., Maves, R., and Colombo, C. (2020). Remdesivir for the treatment of covid-19 — preliminary report. New England Journal of Medicine.

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

Centro-Massagué, A., Ojeda, M. J., Valls, C., Mulero, M., Garcia-Vallvé, S., and Pujadas, G. (2015). Molecular fingerprint similarity search in virtual screening. Methods. 75, 58-63.

Chen, L., Cruz, A., Ramsey, S., Dickson, C. J., and Kurtzman, T. (2019). Hidden. Gschwend, D. A., Good, A. C., and Kuntz, I. D. (2015). Molecular docking towards drug discovery. Journal of Molecular Recognition, 29(2), 175–186.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778.

Hothy, P., Lim, W. S., Embrison, J., Mulham, M., and Lanbery, J. M. (2020). Effect of dexamethasone in hospitalized patients with covid-19. Preliminary report. New England Journal of Medicine.

Hoff, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167.

Kingma, D. P. and Ba, J. A. (2014). A method for stochastic optimization. arxiv 2014. arXiv preprint arXiv:1412.6569.

Led, P. and Callaerts, A. (2017). Protein structure-based drug design: from docking to molecular dynamics. Current Opinion in Structural Biology, 48, 93.

Lee, J., Keum, J., and Nam, H. (2019). Deepcom-dti: Prediction of drug-target interactions via deep learning with convolution on protein sequences. PLoS computational biology, 15(6), e1007129.

Liu, H., Sun, J., Gians, Z., and Zhou, S. (2015). Improving compound–protein interaction prediction by building highly credible negative samples. Briefings in bioinformatics, 16(1), 221–229.

Mehsaei, E. (2020). Covid-19: Low dose steroid cuts death in ventilated patients by one third, trial finds. BMJ (online). 369, m2422.

Min, S., Lee, B., and Youn, S. (2017). Deep learning in bioinformatics. Briefings in bioinformatics, 18(5), 851–869.

Mysinger, M. M., Carchia, M., Irwin, J. J., and Shoichet, B. K. (2012). Directory of useful decoys, enhanced (shul-ds): better ligands and decoys for better benchmarking. Journal of medicinal chemistry, 55(4), 6582–6594.

Nguyen, T., Le, H., and Venkataram, S. (2019). Graphp: prediction of drug-target binding affinity using graph-convolutional networks. BioRxiv, page 684662.

Öztürk, H., Özgür, A., and Özgür, A. (2018). DeepDita: prediction of drug-target binding affinity. Bioinformatics, 34(17), i821–i829.

Öztürk, H., Özgür, A., and Özgür, A. (2019). Widedta: prediction of drug-target binding affinity. arXiv preprint arXiv:1902.04546.

Pazike, A., Gross, S., Masso, F., Lerrer, A., Bradbury, J., Chaanan, G., Killeen, T., Lin, Z., Gmelin, N., Antiga, L., et al. (2019). Pytorch: An imperative style: high-performance deep learning library. In Advances in neural information processing systems, pages 8026–8037.

Pihl, S. M., Mytelka, D. S.,汇总Associate C. T., Pennenger, C. C., Maus, B. H., Lindborg, S. R., and Schacht, A. L. (2010). How to improve r&d productivity: the pharmaceutical industry’s grand challenge. Nature reviews Drug discovery, 9(3), 203–214.

Pruitt, K. D., Tatusova, T., and Maglott, D. R. (2007). Ncbi reference sequences (refseq) a curated non-redundant sequence database of genomes, transcripts and proteins. Nucleic acids research, 35(Database issue), D61-D65.

Ruggiero, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1), 1929–1958.

Tsuchib, M., Tomi, K., and Sese, J. (2019). Compound–protein interaction prediction with end-to-end learning of neural networks for graphs and sequences. Bioinformatics, 35(2), 309–318.

Wang, Z., Zhang, N., Shi, S., Hu, H., Yang, Y., and Lu, H. (2017). Deep-learning-based drug-target interaction prediction. Journal of proteome research, 16(1), 1411–1409.

Yah, J., Chen, C., and Huang, N. (2020). Predicting or pretending: artificial intelligence for protein-ligand interactions lacks of sufficiently large and unbiased datasets. Frontiers in Pharmacology, 11, 69.

Zhang, F., Song, H., Zeng, M., Li, Y., and Li, M. (2019). Protein–protein interaction site prediction through combining local and global features with deep neural networks. Bioinformatics, 36(4).

Zhang, F., Song, H., Zeng, M., Li, Y., and Li, M. (2019). Deepfunc: a deep learning framework for accurate prediction of protein functions from protein sequences and interactions. Proteomics, 19(12), 1909–1918.

Zhang, F., Song, H., Zeng, M., Wu, F., Li, Y., Pan, Y., and Li, M. (2020). A deep learning framework for gene ontology annotations with sequence-and network-based information. IEEE/ACM Transactions on Computational Biology and Bioinformatics.

Zheng, S., Li, Y., Chen, S., Xu, J., and Yang, Y. (2020). Predicting drug–protein interaction using quasi-visual question answering system. Nature Machine Intelligence, 2(2), 134–140.