Molecular Therapy
Nucleic Acids
Original Article

GraphCPIs: A novel graph-based computational model for potential compound-protein interactions

Zhan-Heng Chen,[1,6] Bo-Wei Zhao,[2,6] Jian-Qiang Li,[3] Zhen-Hao Guo,[4] and Zhu-Hong You[5]

[1]Department of Clinical Anesthesiology, Faculty of Anesthesiology, Naval Medical University, Shanghai 200433, China; [2]The Xinjiang Technical Institute of Physics & Chemistry, Chinese Academy of Sciences, Urumqi 830011, China; [3]College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China; [4]Institute of Machine Learning and Systems Biology, School of Electronics and Information Engineering, Tongji University, Caoan Road 4800, Shanghai 201804, China; [5]School of Computer Science, Northwestern Polytechnical University, Xi'an 710129, China

Identifying proteins that interact with drug compounds has been recognized as an important part in the process of drug discovery. Despite extensive efforts that have been invested in predicting compound-protein interactions (CPIs), existing traditional methods still face several challenges. The computer-aided methods can identify high-quality CPI candidates instantaneously. In this research, a novel model is named GraphCPIs, proposed to improve the CPI prediction accuracy. First, we establish the adjacent matrix of entities connected to both drugs and proteins from the collected dataset. Then, the feature representation of nodes could be obtained by using the graph convolutional network and Grarep embedding model. Finally, an extreme gradient boosting (XGBoost) classifier is exploited to identify potential CPIs based on the stacked two kinds of features. The results demonstrate that GraphCPIs achieves the best performance, whose average predictive accuracy rate reaches 90.09%, average area under the receiver operating characteristic curve is 0.9572, and the average area under the precision and recall curve is 0.9621. Moreover, comparative experiments reveal that our method surpasses the state-of-the-art approaches in the field of accuracy and other indicators with the same experimental environment. We believe that the GraphCPIs model will provide valuable insight to discover novel candidate drug-related proteins.

INTRODUCTION

Drugs function usually by interacting with a vast range of compound targets, in which proteins are a principal pattern of targets.[1,2] From *in vivo* experiments, the drug will interact with plasma protein to work when it enters the body. Different drug compounds have different binding rates with proteins. Recently, inferring the relationship between drugs and proteins has become a significant research issue in bioinformatics. The successful identification of compound-protein interactions (CPIs) has been a particularly important step in the incipient stage of drug discovery. However, the traditional biological experiment on CPIs is time consuming and laborious,[3] where it can only discover a single interaction once before the appearance of high-throughput technology. Although these approaches have accumulated considerable data, they are far from complete. Therefore, it is an urgent need to exploit computational techniques to report the most potential drug-related candidates for further prediction of CPIs, which can save time and cost of traditional wet-lab experiments and accelerate the drug exploitation process.

Recently, the prediction of CPIs has been regarded as a binary classification problem,[4] in which the construction of the dataset is an important component. Over the past decade, there has been the emergence of numerous drug-related databases along with the development of high-throughput technology, such as DrugBank,[5] PubChem,[6] TTD,[7] ChEMBL,[8] and BindingDB.[9] The abundant data provide valuable insight into studying CPIs, contributing to the birth of a state-of-the-art (SOTA) calculation model for inferring CPIs depending on machine learning. Based on these reliable data sources, a variety of biological information (e.g., protein homology, protein function information, protein sequence, drug chemical structure, molecular fingerprint, and so on) can be fused, and supervised and semi-supervised learning methods can be applied to effectively predict potential CPIs. Lee et al.[10] performed a convolutional neural network (CNN) on amino acid sequences to acquire regional residue features of proteins, so as to realize the CPIs' prediction and binding sites of proteins. Mahmud et al.[11] used three descriptors and the molecular substructure fingerprint to describe the features from the amino acid sequence and compound chemical structure, which was input into an iDTi-CSsmoteB model for the identification of drug-protein interactions. Bleakley et al.[12] applied the bipartite local

Received 15 November 2022; accepted 28 April 2023;
https://doi.org/10.1016/j.omtn.2023.04.030.

[6]These authors contributed equally

Correspondence: Zhan-Heng Chen, Department of Clinical Anesthesiology, Faculty of Anesthesiology, Naval Medical University, Shanghai 200433, China.
E-mail: chenzhanheng17@mails.ucas.ac.cn
Correspondence: Jian-Qiang Li, College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China.
E-mail: lijq@szu.edu.cn
Correspondence: Zhu-Hong You, School of Computer Science, Northwestern Polytechnical University, Xi'an 710129, China.
E-mail: zhuhongyou@nwpu.edu.cn

Table 1. The measurement results of GraphCPIs using 5-fold cross-validation on benchmark dataset

Fold	Acc. (%)	TPR (%)	TNR (%)	PPV (%)	MCC
1	90.44	87.41	93.46	93.04	0.8103
2	90.37	86.80	93.95	93.48	0.8095
3	89.93	85.70	94.17	93.63	0.8016
4	89.45	85.39	93.51	92.94	0.7916
5	90.26	85.48	95.04	94.52	0.8090
Average	**90.09 ± 0.41**	**86.16 ± 0.90**	**94.03 ± 0.64**	**93.52 ± 0.63**	**0.8044 ± 0.0080**

models to predict unknown drug-protein interactions in humans. This method only relies on the chemical structure of compounds and amino acid sequence information, which can screen drug candidate molecules and candidate proteins on a large scale.

In addition, graph- and network-based approaches are also being upgraded. For instance, DINIES is an online system established by Yamanishi et al.[13] that is based on a framework of supervised network inference employed to predict unknown CPIs from many types of biological data. According to the quasi-visual question-answering mode, Zheng et al.[14] developed an end-to-end deep learning model to recognize CPIs based on the molecular liner notation of drugs and a 2-dimensional distance map from the monomer structure of proteins by taking advantage of dynamic attentive CNN. Wu et al.[15] constructed a learnable drug-protein interaction network by using a graph neural network to dig up the network-level representation from compounds and amino acids. A unified framework was established by Ye et al.[16] based on a knowledge graph and recommendation system. Zhao et al.[17] inferred the interaction between drugs and proteins by using large-scale graph representation learning to obtain the different types of structural information from the organized interactions network. Chen et al.[18] constructed a multi-association graph by combining the relationships among five types of molecules; they took advantage of the representation learning method to obtain behavior features that input into the various classifiers to predict drug CPIs.

Based on the above observations, we proposed a graph- and network-based deep learning framework, termed GraphCPIs, to accomplish a classification task on CPIs' predictions. A core insight of our work is that the various kinds of features are described from different methods, in particular the graph-based features and learning-based features. GraphCPIs mainly differs from other previous studies in the following three factors: (1) the network embedding learning method is borrowed to represent the features between drug molecules and proteins; (2) by combining graph and network embedding method to obtain the feature vectors of pairwise CPIs, GraphCPIs takes graph representations as inputs to learn structural and sequential information for proteins and drugs; (3) XGBoost classifier is introduced to achieve high performance with the prediction of CPIs. Comprehensive experiments on different feature representations and various kinds of classifiers indicate that GraphCPIs surpasses all other SOTA approaches in five evaluation criteria, average area under the receiver operating characteristic (AU-ROC), and average area under the precision and recall (AU-PR). The overall model structure of our proposed GraphCPIs is shown in Figure S1.

RESULTS

Performance of GraphCPIs using 5-fold cross-validation

Here, we propose a graph recombination method, called GraphCPIs, to predict drug CPIs. To verify and access the performance of the proposed method, 5-fold cross-validation is utilized. More specifically, we randomly split the benchmark dataset into five roughly equivalent pieces, four of which are for training; the fifth part is employed for testing. This step is repeated five times, and a different test set is selected each time. The final average results from these runs can explain the stability of the proposed model.

Focused on binary classification-based CPIs detection studies, the true labels and predicted labels are calculated from positive and negative samples that make up a confusion matrix (CM, also called a table of confusion). In this study, five information metrics calculated on CM are introduced to summarize the predictive values on the benchmark dataset, i.e., Acc. (accuracy, a measure of observational error), TPR (true positive rate, or sensitivity, or recall), TNR (true negative rate, or specificity), PPV (positive predictive value, or precision), and MCC (Matthews correlation coefficient).[19] These predictive values are shown in Table 1.

On the other hand, the AU-ROC and AU-PR are introduced to further showcase the predictive capability of the proposed model on binary classification. The AU-ROC is a chart indicating the measurement of the machine learning method across all possible thresholds. AU-ROC includes two parameters: sensitivity and 1-specificity. The closer the AU-ROC value is to 1, the better effect of the classification there will be. As shown in Figure S2, each fold value of AU-ROC is very stable, and the average AUC value is 0.9572. The AU-PR indicates the relation between precision and recall, which profiles the description of data distribution. It is more informative about accessing the overall performance of the classification model.[20] AU-PR plots two parameters: precision and recall. The higher the AU-PR value is, the higher is the correct rate. As shown in Figure S2, each fold value of AU-PR is very stable, and its average value is 0.9621.

Comparing the performance of various features on the same classifier

In order to verify that the GraphCPIs method can obtain more useful feature information, we compared various features with our method by taking advantage of different feature extraction methods. In this work, we combined the graph convolutional network and network embedding method for improving the prediction performance. To appraise the impact of the GraphCPIs model, we tested the outcome of three kinds of feature construction strategies: (1) sequence-based features analysis, incorporating drug molecular fingerprint feature and amino acids sequence feature of protein; (2) graph convolutional

Table 2. The performance results of sequence-based features using 5-fold cross-validation on benchmark dataset

Fold	Acc. (%)	TPR (%)	TNR (%)	PPV (%)	MCC
1	85.68	85.35	86.01	85.92	0.7136
2	85.26	85.75	84.78	84.93	0.7053
3	86.69	85.61	87.76	87.49	0.7339
4	85.81	84.82	86.80	86.53	0.7164
5	86.03	84.12	87.94	87.46	0.7211
Average	**85.89 ± 0.53**	**85.13 ± 0.67**	**86.66 ± 1.31**	**86.47 ± 1.08**	**0.7181 ± 0.0011**

Table 3. The performance results of GCN-based features using 5-fold cross-validation on benchmark dataset

Fold	Acc. (%)	TPR (%)	TNR (%)	PPV (%)	MCC
1	86.32	85.00	87.63	87.30	0.7266
2	86.18	84.43	87.94	87.50	0.7241
3	85.35	82.41	88.29	87.56	0.7082
4	85.42	82.32	88.51	87.75	0.7097
5	85.55	82.76	88.33	87.65	0.7121
Average	**85.76 ± 0.45**	**83.38 ± 1.24**	**88.14 ± 0.35**	**87.55 ± 0.17**	**0.7161 ± 0.0086**

network (GCN)-based features; (3) network embedding (NE)-based features. The experiments are carried out by adopting 5-fold cross-validation, and the performances are listed in Tables 2, 3, and 4. These data prove that the hybrid feature we consider is useful and effective.

In each fold from Tables 2, 3, and 4, we obtained TPRs and 1-TNRs. Then, we plotted the corresponding ROC curves based on these data and calculated the AU-ROC values as a comprehensive evaluation reference of different features. For the same reason, PPV and recall can be applied to compute the AU-PR values. As shown in Figure 1, we additionally explored the combination of two kinds of features with three other single features tested on the standard dataset, whose results indicate that their performances are outstanding. We show that GraphCPIs achieved a mean AU-ROC of 0.9572 under 5-fold cross-validation and a mean AU-PR of 0.9621 upon 5-fold cross-validation. Meanwhile, the mean AU-ROC values of the other three types of features (GCN-based, sequence-based, Grarep-based) are 0.9266, 0.9250, and 0.9566, respectively. The mean AU-PR values of the other three types of features (GCN-based, sequence-based, Grarep-based) are 0.9295, 0.9268, and 0.9619, respectively.

Comparing the performance of different classifiers

To further support the results listed above, 5-fold cross-validation was utilized to test various more advanced classifiers. Simultaneously, we compared the XGBoost classifier with the other three commonly used classifiers, including classification and regression tree (CART classifier),[21,22] Gaussian naive Bayes (GNB classifier),[23] and support vector machine (SVM). All these classifiers are realized on the standard dataset and the same features by using 5-fold cross-validation.

In these comparison results from Tables S1–S3 and Figure 2, we carry out five times the experiments for each classifier using the same feature representation method on the standard dataset. The XGBoost classifier obtained the most optimal result, reaching an average Acc. of 90.09, which is better than the CART, GNB, and SVM classifiers by 7.58%, 20.89%, and 0.22%, respectively. XGBoost received the highest average MCC with 0.8044, which is 0.1540 more than CART, 0.4082 more than GNB, and 0.0062 slightly more than SVM. In short, the results of XGBoost are the best on five evaluation criteria, and the prediction performance is quite sensitive to the different classifier choices. Meanwhile, as listed in Table 5, our work is also compared with the previous related work proposed by Zhao et al.,[17] and the overall performance of our proposed algorithm is slightly better.

Moreover, we further evaluate the performance of different classifiers, the AU-ROC, and the AU-PR as in Figure 3. It is obvious that XGB received the most competitive results, which is related to the following factors: (1) for the CART classifier, it is easy to ignore the correlation between features when only using one field for classification; (2) for the GNB classifier, the classification effect mainly relies on the correlation between features, where the smaller the attribute correlation is, the better is the performance of GNB; (3) the XGB classifier supports column sampling, which can reduce over-fitting and calculation and increase the generalization performance of the model.

Case study

In the case study, all known drug-target interactions in the benchmark dataset are first taken as positive samples to compose the training dataset, and they are collected from DrugBank V5.0 database (Release 2019). In this regard, unknown CTIs refers to novel CTIs that are not found in DrugBank V5.1.10 database (Release 2023) but are predicted by GraphCPIs model. We then verify these unknown CTIs in the latest version of DrugBank, i.e., V5.1.10. which is the latest release of DrugBank Online (version 5.1.10, released 01-04-2023, https://go.drugbank.com/). In other words, these verified CTIs do not exist in DrugBank V5.0 but were later added into DrugBank V5.1.10 due to the update of this database.

Table 4. The performance results of NE-based features using 5-fold cross-validation on benchmark dataset

Fold	Acc. (%)	TPR (%)	TNR (%)	PPV (%)	MCC
1	90.26	86.58	93.95	93.47	0.8075
2	90.18	86.49	93.86	93.37	0.8057
3	89.91	85.31	94.52	93.96	0.8017
4	89.17	84.74	93.60	92.97	0.7864
5	89.89	84.78	95.00	94.43	0.8020
Average	**89.88 ± 0.43**	**85.58 ± 0.90**	**94.19 ± 0.57**	**93.64 ± 0.57**	**0.8007 ± 0.0083**

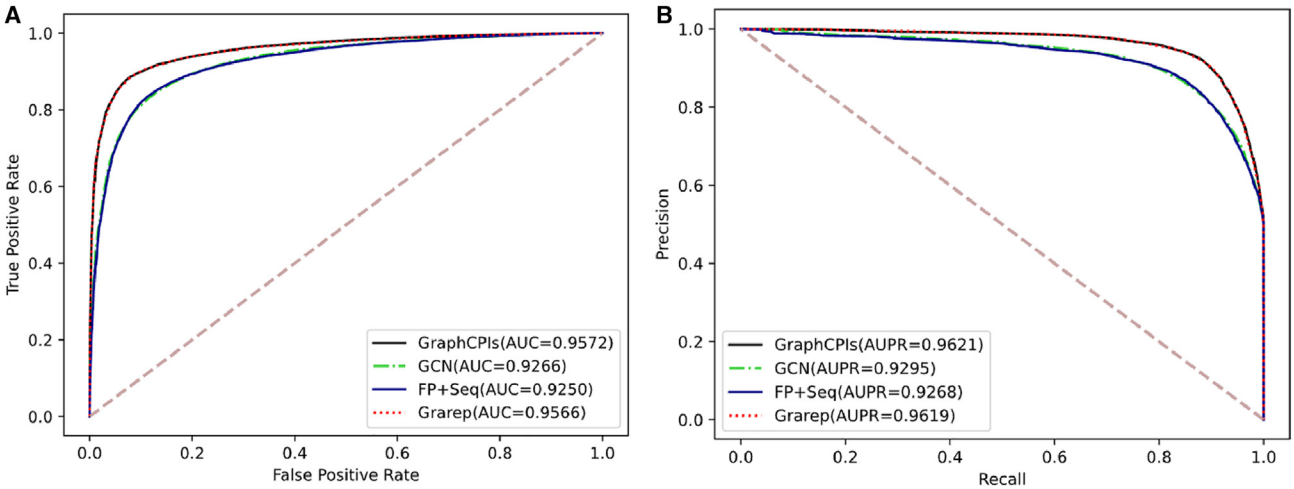

Figure 1. Performances comparison on the prediction of XGBoost that applies different features in 5-fold cross-validation

(A) The AU-ROC values based on different features. (B) The AU-PR values based on different features.

In order to bring stronger evidence for our model, all confirmed CPIs from the collected dataset are trained by GraphCPIs, and then the prediction results of ten drugs and corresponding related proteins are as shown in Table S4. Among them, it is shown that there were seven proteins of corresponding drugs recognized by GraphCPIs that can be confirmed from the DrugBank. The rest of the unconfirmed proteins could be potential ones, expected to be checked by medical experts. Moreover, we hope this work and its application will provide broad prospects for the discovery of new candidate drug-related targets.

DISCUSSION

Identifying interactions between drug compounds and target proteins is still a fundamental challenge, and the relevant prediction model is also not well explored. In this study, a computational model termed GraphCPIs is developed based on GCN and Grarep method to detect a potential relationship between drug molecules and proteins. Specifically, the network graph is constructed by the known relationships between drugs and proteins in the collected dataset. We then trained GCN and Grarep models to learn efficient vector representation for these nodes in the above graph. Finally, two combined features are fed into the XGBoost classifier to complete the task of CPIs classification. GraphCPIs can obtain an AUC value of 0.9572 and an AU-PR of 0.9621 that surpasses all other SOTA models.

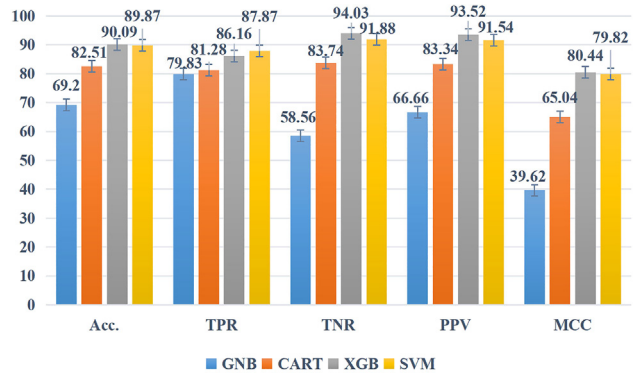

Figure 2. Performances comparison of various classifiers based on GraphCPIs features

In addition, a series of comparison experiments are conducted on the benchmark dataset we gathered, and a comprehensive analysis is also made of the predicted results. At first, the various features are introduced for comparative experiments based on the same classifier (XGBoost), e.g., sequence-based feature, GCN-based feature, Grarep-based feature, and GraphCPIs-based feature. Then, the XGBoost classifier is compared with the SOTA classifiers based on the same feature (GraphCPIs-based feature), e.g., CART classifier, GNB classifier, and SVM classifier. Finally, the proposed GraphCPIs model is also compared with the previous related work. All the comparison performance results indicate the practicability and effectiveness of the GraphCPIs model in the task of detecting CPIs.

MATERIALS AND METHODS

Benchmark data collection

In this study, the benchmark dataset mainly stems from DrugBank5.0,[5] which is a free-to-access, comprehensive, online database. This database integrates a great deal of information on drug compounds, drug-related proteins, and drug-protein interaction pairs, concentrating primarily on the associations of proteins regarded as targeted with drug-like molecules. We simply withdrew 11,396 known CPIs from the DrugBank, including 635 protein targets and 984 drug molecules. These known CPIs are regarded as positive samples, which are a significant part of the benchmark dataset. The 11,396 negative samples were randomly extracted from the remaining uncertain interaction data (635 × 984 − 11,396 = 613,444). Finally, the

Table 5. Compared GraphCPIs with the previous related work

Model	Acc. (%)	TPR (%)	TNR (%)	PPV (%)	MCC	AUC
Zhao et al.[17]	88.64	83.67	93.61	92.90	0.7766	0.9455
GraphCPIs	90.09	**86.16**	**94.03**	**93.52**	**0.8044**	**0.9572**

generation of eventual benchmark dataset integrity was from positive and negative samples, whose size is 22,792.

Feature representation methods

Drug Morgan fingerprint

Molecular fingerprint is the abstract expression of compounds, which can encode drugs into lots of vectors. Morgan fingerprint is one of the most popular molecular fingerprints, also known as extended-connectivity fingerprints (ECFPs).[24] It is a novel class of circular fingerprint with plenty of helpful qualities, which also belongs to the topological fingerprint for molecular characterization. This method is derived using a variant of the standard Morgan algorithm,[25] which is developed for addressing the molecular isomorphism problem and aims to identify the substructures in molecules without the way of atom-relating order. Here, the Morgan fingerprints of drugs are calculated by the RDKit tool in python.[26]

Take amoxicillin (used to treat certain infections caused by bacteria) as a brief example; all the substructures in amoxicillin can be obtained with a radius of 2. Hence, each bit on the molecular fingerprint corresponds to an atom substructure, and then the structural features of the molecule will be extracted to generate a bit vector. As shown in Figure S3, we only give partial atom substructures here.

Protein feature representation

Protein sequences are mainly selected from the STRING database.[27] We all know that all proteins are composed of 20 amino acids, which can be divided into four groups, namely Asp and Glu; Arg, Lys, and His; Gly, Ser, Thr, Cys, Asn, Gln, and Tyr; and Ala, Val, Leu, Ile, Met, Phe, Trp, and Pro. Here, k-mer ($k = 3$) is introduced to transform every protein to a 64-dimensional eigenvector by counting the number of occurrences with every k amino acids from the entire sequence.

Graph representation learning model

The graph representation learning model (Grarep)[28,29] is concerned with learning the potential vector expression of nodes in the graph, which can capture the global structural information. In representation learning, Grarep is employed to better distinguish the neighbors with different orders of nodes, and it can also be extended to the neighbors with any orders.

Grarep is an algorithm that maps the k-order information of nodes to diverse sub-spaces. One can capture the k-step relational information (k can take different values) of nodes by manipulating different global probability transition matrices A^k. Then, the global expression is obtained by integrating this k-order local information. The description of k-step information is as shown in Figure S4.

The k-step probability transition matrix is defined as follows:

$$A^k = \underbrace{A\cdots\cdots A}_{k}, A = D^{-1}H, \quad \text{(Equation 1)}$$

where A_{ij}^k indicates the probability that i skips to j in k steps. D denotes the degree matrix of a node, and H is an adjacency matrix, which is the edge from i to j. Thus, the k-step transition probability from current vertex m to context vertex n will be described as follows:

$$p_k(n|m) = A_{m,n}^k \quad \text{(Equation 2)}$$

Here, the objective function is defined from noise contrastive estimation.[30] Following a similar discussion presented in the work of Levy et al.,[31] the k-step loss function is defined as follows:

$$L_k = \sum_{m\in V} L_k(m), \quad \text{(Equation 3)}$$

where

$$L_k(m) = \left(\sum_{c\in V} p_k(n|m)\log\sigma(\vec{m}\cdot\vec{n})\right) + \lambda E_{n'\sim p_k(V)}[\log\sigma(-\vec{m}\cdot\vec{n})] \quad \text{(Equation 4)}$$

Here, $V = \{v_1, v_2, \ldots\ldots, v_n\}$ is the set of vertices with every V indicating one object (Nv is the number of vertices in the graph), $\sigma(x) = (1 + e^{-x})^{-1}$ represents a *sigmoid* function, λ denotes the hyper-parameter representing the count of negative entities, and $p_k(V)$ indicates a distribution of context vertex $n\in V$ in the path of k-step. The expectation $\boldsymbol{E}$ could be briefly defined as follows:

$$\begin{aligned} &E_{n'\sim p_k(V)}[\log\sigma(-\vec{m}\cdot\vec{n})] \\ &= p_k(n)\cdot\log\sigma(-\vec{m}\cdot\vec{n}) \\ &+ \sum_{n'\in V\setminus\{n\}} p_k(n')\cdot\log\sigma(-\vec{m}\cdot\vec{n}'), \end{aligned} \quad \text{(Equation 5)}$$

where

$$\begin{aligned} p_k(n) &= \sum_{m'} q(m')p_k(n|m') \\ &= \frac{1}{N_V}\sum_{m'} A_{m',n}^k \end{aligned} \quad \text{(Equation 6)}$$

Note that here $q(m') = 1/Nv$ is the probability of choosing m' as the first node, which corresponds to a uniform distribution. Then,

$$L_k(m,n) = \left(A_{m,n}^k\cdot\log\sigma(\vec{m}\cdot\vec{n})\right) + \frac{\lambda}{N_V}\sum_{m'} A_{m',n}^k\cdot\log\sigma(-\vec{m}\cdot\vec{n}) \quad \text{(Equation 7)}$$

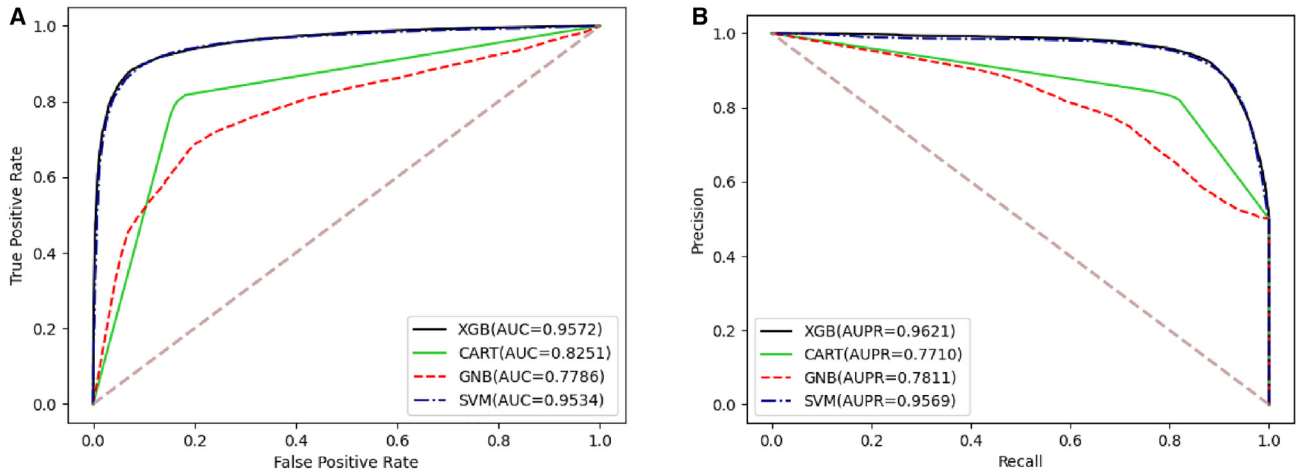

Figure 3. The AU-ROC (A) and AU-PR (B) values are based on different classifiers

Therefore, we find a matrix Y that encodes the relationship between all nodes in the graph.

$$Y = \vec{m} \cdot \vec{n} = \log\left(\frac{A^k_{m,n}}{\sum_{m'} A^k_{m',n}}\right) - \log(\beta), \beta = \frac{1}{N_V} \quad \text{(Equation 8)}$$

To lessen the effect of error and form a new matrix X, all the negative values in Y are replaced with 0. Thereby, the above-mentioned loss essentially belongs to the matrix factorization problem (singular value decomposition factorization[32,33]).

$$X^k_{i,j} = \max\left(Y^k_{i,j}, 0\right) \quad \text{(Equation 9)}$$

$$X^k \approx X^k_d = U^k_d \Sigma^k_d \left(V^k_d\right)^T = M^k N^k \quad \text{(Equation 10)}$$

Where, U and V are orthonormal matrices, $\sum$ denotes the diagonal matrix, and d is the dimension of the final eigenvector. The network representation of current node M and context node N in the graph are obtained as follows:

$$M^k = U^k_d \left(\Sigma^k_d\right)^{\frac{1}{2}} \quad \text{(Equation 11)}$$

$$N^k = \left(\Sigma^k_d\right)^{\frac{1}{2}} V^{k^T}_d \quad \text{(Equation 12)}$$

Subsequently, the feature vectors of all the drug molecules and proteins in the benchmark dataset can be obtained by the Grarep NE method, which includes the global structure information in the graph.

Construction of GraphCPIs model

We construct a GraphCPIs model for predicting drug CPIs by combining two kinds of graph-related methods. GraphCPIs contains two main components: (1) GCN[34] and (2) Grarep NE method. In the first step, all the related heterogeneous information from the collected dataset is exploited to build a graph network, where each type of drug and protein is regarded as a node, and the interactions between them are considered an edge. In the second step, we constructed a GCN network to capture the local similarity between pairs of drugs and proteins in the graph. In the third step, the Grarep embedding method is applied to learn in-depth embedding vectors for known entities and their relations. Finally, two combined features are fed into the XGBoost classifier to predict the potential CPIs.

Extreme gradient boosting ensemble methods

The goal behind ensemble methods is to combine a sequence of weak classifiers into a meta-classifier that has a better generalization performance than every single classifier.[35] Extreme gradient boosting (XGBoost) ensemble classifier was first proposed by Friedman in 2001,[36] and it is a widely used model in the field of machine learning.[37] XGBoost is an open source framework for gradient boosting created by Chen et al.,[38] and it is available in popular languages such as Julia, R, Python, and so on.

The learning objective is to find the difference value (D-value) of the second-order Taylor expansion of the loss function, which is equivalent to employing the Newton method to optimize and approximate the minimum value of the loss function.

$$L(\theta) = \sum_{i=1}^{n} l\left(p_i, t_i\right) + \sum_{j}^{J} \Omega\left(f_j\right), \quad \text{(Equation 13)}$$

where the former part is the training loss of gradient boosting algorithm, n denotes the quantity of training instances, l represents a differentiable convex loss function, p_i is the predictive value of

training entities, and t_i is the true value of training entities. The latter part annotates the regularization term that represents the complexity of a model, where each f_j corresponds to an independent tree structure and leaf weights.

$$\Omega(f) = \gamma D + \frac{1}{2}\lambda\|V\|^2 = \gamma D + \frac{1}{2}\lambda\sum_{j=1}^{D} V_j^2, \quad \text{(Equation 14)}$$

where γ and λ are manually set parameters to avoid an over-fitting problem, D is the number of leaf nodes, and V denotes a vector generated by all leaf nodes from the decision tree.

DATA AVAILABILITY

The dataset presented in the study is available at https://github.com/gitlearning518/GraphCPIs. The source code of GraphCPIs is available at https://github.com/gitlearning518/GraphCPIs.

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.omtn.2023.04.030.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China under grants 62002297, 62073225, 61902342, U2013201, and 61836005, the Natural Science Foundation of Guangdong Province-Outstanding Youth Program under grant 2019B151502018, and the Guangxi Science and Technology Base and Talent Special Project under grant 2021AC19394. The authors would like to thank all the guest editors and anonymous reviewers for their constructive advice.

AUTHOR CONTRIBUTIONS

Z.-H.C.: conceptualization, methodology, software, data curation, funding acquisition, and writing – original draft preparation. B.-W.Z.: visualization, investigation, and data collection. J.-Q.L.: supervision and funding acquisition. Z.-H.G.: data collection, software, and reviewing. Z.-H.Y.: validation, and writing – reviewing and editing. The corresponding author is responsible for ensuring that the descriptions are accurate and agreed to by all authors.

DECLARATION OF INTERESTS

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

REFERENCES

1. Huang, K., Xiao, C., Glass, L.M., and Sun, J. (2021). MolTrans: molecular Interaction Transformer for drug–target interaction prediction. Bioinformatics *37*, 830–836.
2. Sun, C., Cao, Y., Wei, J.M., and Liu, J. (2021). Autoencoder-based drug–target interaction prediction by preserving the consistency of chemical properties and functions of drugs. Bioinformatics *37*, 3618–3625.
3. Bagherian, M., Sabeti, E., Wang, K., Sartor, M.A., Nikolovska-Coleska, Z., and Najarian, K. (2021). Machine learning approaches and databases for prediction of drug–target interaction: a survey paper. Brief. Bioinform. *22*, 247–269.
4. Pahikkala, T., Airola, A., Pietilä, S., Shakyawar, S., Szwajda, A., Tang, J., and Aittokallio, T. (2015). Toward more realistic drug–target interaction predictions. Brief. Bioinform. *16*, 325–337.
5. Wishart, D.S., Feunang, Y.D., Guo, A.C., Lo, E.J., Marcu, A., Grant, J.R., Sajed, T., Johnson, D., Li, C., Sayeeda, Z., et al. (2018). DrugBank 5.0: a major update to the DrugBank database for 2018. Nucleic Acids Res. *46*, D1074–D1082.
6. Kim, S., Chen, J., Cheng, T., Gindulyte, A., He, J., He, S., Li, Q., Shoemaker, B.A., Thiessen, P.A., Yu, B., et al. (2019). PubChem 2019 update: improved access to chemical data. Nucleic Acids Res. *47*, D1102–D1109.
7. Wang, Y., Zhang, S., Li, F., Zhou, Y., Zhang, Y., Wang, Z., Zhang, R., Zhu, J., Ren, Y., Tan, Y., et al. (2020). Therapeutic target database 2020: enriched resource for facilitating research and early development of targeted therapeutics. Nucleic Acids Res. *48*, D1031–D1041.
8. Mendez, D., Gaulton, A., Bento, A.P., Chambers, J., De Veij, M., Félix, E., Magariños, M.P., Mosquera, J.F., Mutowo, P., Nowotka, M., et al. (2019). ChEMBL: towards direct deposition of bioassay data. Nucleic Acids Res. *47*, D930–D940.
9. Gilson, M.K., Liu, T., Baitaluk, M., Nicola, G., Hwang, L., and Chong, J. (2016). BindingDB in 2015: a public database for medicinal chemistry, computational chemistry and systems pharmacology. Nucleic Acids Res. *44*, D1045–D1053.
10. Lee, I., Keum, J., and Nam, H. (2019). DeepConv-DTI: prediction of drug-target interactions via deep learning with convolution on protein sequences. PLoS Comput. Biol. *15*, e1007129.
11. Mahmud, S.M.H., Chen, W., Jahan, H., Liu, Y., Sujan, N.I., and Ahmed, S. (2019). iDTi-CSsmoteB: identification of drug–target interaction based on drug chemical structure and protein sequence using XGBoost with over-sampling technique SMOTE. IEEE Access *7*, 48699–48714.
12. Bleakley, K., and Yamanishi, Y. (2009). Supervised prediction of drug–target interactions using bipartite local models. Bioinformatics *25*, 2397–2403.
13. Yamanishi, Y., Kotera, M., Moriya, Y., Sawada, R., Kanehisa, M., and Goto, S. (2014). DINIES: drug–target interaction network inference engine based on supervised analysis. Nucleic Acids Res. *42*, W39–W45.
14. Zheng, S., Li, Y., Chen, S., Xu, J., and Yang, Y. (2020). Predicting drug–protein interaction using quasi-visual question answering system. Nat. Mach. Intell. *2*, 134–140.
15. Wu, Y., Gao, M., Zeng, M., Zhang, J., and Li, M. (2022). BridgeDPI: a novel Graph Neural Network for predicting drug–protein interactions. Bioinformatics *38*, 2571–2578.
16. Ye, Q., Hsieh, C.Y., Yang, Z., Kang, Y., Chen, J., Cao, D., He, S., and Hou, T. (2021). A unified drug–target interaction prediction framework based on knowledge graph and recommendation system. Nat. Commun. *12*, 6775.
17. Zhao, B.-W., You, Z.-H., Hu, L., Guo, Z.H., Wang, L., Chen, Z.H., and Wong, L. (2021). A novel method to predict drug-target interactions based on large-scale graph representation learning. Cancers *13*, 2111.
18. Chen, Z.H., You, Z.H., Guo, Z.H., Yi, H.C., Luo, G.X., and Wang, Y.B. (2020). Prediction of drug–target interactions from multi-molecular network based on deep walk embedding model. Front. Bioeng. Biotechnol. *8*, 338.
19. Luque, A., Carrasco, A., Martín, A., and de Las Heras, A. (2019). The impact of class imbalance in classification performance metrics based on the binary confusion matrix. Pattern Recognit *91*, 216–231.
20. Saito, T., and Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PLoS One *10*, e0118432.
21. Lewis, R.J. (2000). An introduction to classification and regression tree (CART) analysis. In Annual Meeting of the Society for Academic Emergency Medicine in San Francisco, California, *14* (Department of Emergency Medicine Harbor-UCLA Medical Center Torrance).
22. Javed Mehedi Shamrat, F., Ranjan, R., Hasib, K.M., Yadav, A., and Siddique, A.H. (2022). Performance evaluation among ID3, C4. 5, and CART decision tree algorithm. In Pervasive Computing and Social Networking. Proceedings of ICPCSN 2021 (Springer), pp. 127–142.

23. Jayachitra, S., and Prasanth, A. (2021). Multi-feature analysis for automated brain stroke classification using weighted Gaussian naïve Bayes classifier. J. Circ. Syst. Comput. *30*, 2150178.

24. Rogers, D., and Hahn, M. (2010). Extended-connectivity fingerprints. J. Chem. Inf. Model. *50*, 742–754.

25. Morgan, H.L. (1965). The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service. J. Chem. Doc. *5*, 107–113.

26. Landrum, G. (2013). Rdkit documentation. Release *1*, 4.

27. Szklarczyk, D., Morris, J.H., Cook, H., Kuhn, M., Wyder, S., Simonovic, M., Santos, A., Doncheva, N.T., Roth, A., Bork, P., et al. (2017). The STRING database in 2017: quality-controlled protein–protein association networks, made broadly accessible. Nucleic Acids Res. *45*, D362–D368.

28. Cao, S., Lu, W., and Xu, Q. (2015). Grarep: learning graph representations with global structural information. In Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pp. 891–900.

29. Ji, B.Y., You, Z.H., Cheng, L., Zhou, J.R., Alghazzawi, D., and Li, L.,P. (2020). Predicting miRNA-disease association from heterogeneous information network with GraRep embedding model. Sci. Rep. *10*, 6658.

30. Gutmann, M.U., and Hyvärinen, A. (2012). Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. J. Mach. Learn. Res. *13*, 307–361.

31. Levy, O., and Goldberg, Y. (2014). Neural word embedding as implicit matrix factorization. Adv. Neural Inf. Process. Syst. *27*.

32. Andrews, H., and Patterson, C. (1976). Singular value decomposition (SVD) image coding. IEEE Trans. Commun. *24*, 425–432.

33. Li, H., Liu, T., Wu, X., and Chen, Q. (2021). A bearing fault diagnosis method based on enhanced singular value decomposition. IEEE Trans. Industr. Inform. *17*, 3220–3230.

34. Welling, M., and Kipf, T.N. (2016). Semi-supervised Classification with Graph Convolutional Networks. Preprint at arXiv. https://doi.org/10.48550/arXiv.1609.02907.

35. Sagi, O., and Rokach, L. (2018). Ensemble learning: a survey. Wiley Interdiscip. Rev. Data Min. Knowl. Discov. *8*, e1249.

36. Friedman, J.H. (2001). Greedy function approximation: a gradient boosting machine. Ann. Stat. *29*, 1189–1232.

37. Yu, B., Qiu, W., Chen, C., Ma, A., Jiang, J., Zhou, H., and Ma, Q. (2020). SubMito-XGBoost: predicting protein submitochondrial localization by fusing multiple feature information and eXtreme gradient boosting. Bioinformatics *36*, 1074–1081.

38. Chen, T., and Guestrin, C. (2016). Xgboost: a scalable tree boosting system. In Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, pp. 785–794.