

FMGNN: A Method to Predict Compound-Protein Interaction With Pharmacophore Features and Physicochemical Properties of Amino Acids

Chunyan Tang , Cheng Zhong , Mian Wang, and Fengfeng Zhou 

Abstract—Identifying interactions between compounds and proteins is an essential task in drug discovery. To recommend compounds as new drug candidates, applying the computational approaches has a lower cost than conducting the wet-lab experiments. Machine learning-based methods, especially deep learning-based methods, have advantages in learning complex feature interactions between compounds and proteins. However, deep learning models will over-generalize and lead to the problem of predicting less relevant compound-protein pairs when the compound-protein feature interactions are high-dimensional sparse. This problem can be overcome by learning both low-order and high-order feature interactions. In this paper, we propose a novel hybrid model with Factorization Machines and Graph Neural Network called FMGNN to extract the low-order and high-order features, respectively. Then, we design a compound-protein interactions (CPIs) prediction method with pharmacophore features of compound and physicochemical properties of amino acids. The pharmacophore features can ensure that the prediction results much more fit the expectation of biological experiment and the physicochemical properties of amino acids are loaded into the embedding layer to improve the convergence speed and accuracy of protein feature learning. The experimental results on several datasets, especially on an imbalanced large-scale dataset, showed that our proposed method outperforms other existing methods for CPI prediction. The western blot experiment results on wogonin and its candidate target proteins also showed that our proposed method is effective and accurate for finding target proteins. The computer program of implementing the model FMGNN is available at <https://github.com/tcygxu2021/FMGNN>.

Index Terms—Compound protein interaction, Deep learning, Graph Neural Network, Factorization Machines, Pharmacophore features

1 INTRODUCTION

THE identification of compound-protein interactions (CPIs) is of extraordinary significance to modern drug discovery in terms of suggesting new drug candidates and repositioning old drugs. The biological assays for CPIs identification like high-throughput screening assays, are still extremely experimental costly. To reduce the experimental

cost, computational methods for identifying potential CPIs were proposed in the past decade [1], [2], [3].

To identify potential CPIs, a variety of machine learning based predicting algorithms have been proposed since 2008. Most of machine learning based CPIs prediction methods treat the CPIs prediction problem as binary classification task, in which its goal is to determine whether a compound-protein pair interacts. The CPIs prediction procedure mainly consists of generating feature vectors, training model with known CPIs, and predicting unknown compound-protein pairs on the trained model. Yamanishi *et al.* [4] proposed a bipartite network by integrating the chemical and genomic features into a pharmacology feature space, and applied a kernel regression method to predict CPIs. Bleakley *et al.* [5] presented a supervised bipartite local model called BLM using support vector machines (SVM) classifier to predict drug and target sets respectively. Laarhoven *et al.* [6] constructed a Gaussian interaction profile (GIP) kernel to capture the topological features in CPIs network. The Matrix Factorization (MF) based methods were introduced by decomposing the interaction feature vectors into drug latent factors and target latent factors to predict potential CPIs [7], [8], [9]. By treating the CPIs as a network link prediction problem, Chen *et al.* [10] developed a network-based random walk model with restart on heterogeneous networks (NRWRH) to predict potential CPIs. Tang *et al.* [11] proposed a method called MDMHN to predict hidden or missing CPIs on a heterogeneous network by transforming a compound-protein interaction pairs prediction problem to a matrix denoising problem.

- Chunyan Tang is with the School of Computer Science and Engineering, South China University of Technology, Guangzhou, Guangdong 510006, China, and with the School of Computer, Electronics and Information, Guangxi University, Nanning, Guangxi 530004, China, and also with the Key Laboratory of parallel distributed Computing Technology in Guangxi Universities, Guangxi University, Nanning, Guangxi 530004, China. E-mail: tangchunyan@gxu.edu.cn.
- Cheng Zhong is with the School of Computer, Electronics and Information, Guangxi University, Nanning, Guangxi 530004, China, and also with the Key Laboratory of parallel distributed Computing Technology in Guangxi Universities, Guangxi University, Nanning, Guangxi 530004, China. E-mail: chzhong@gxu.edu.cn.
- Mian Wang is with the College of Life Science and Technology, Guangxi University, Nanning, Guangxi 530004, China. E-mail: mianwang@gxu.edu.cn.
- Fengfeng Zhou is with the College of Computer Science and Technology, Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun, Jilin 130012, China. E-mail: fengfengzhou@gmail.com.

Manuscript received 28 Sept. 2021; revised 24 Mar. 2022; accepted 30 Apr. 2022. Date of publication 3 May 2022; date of current version 3 Apr. 2023.

This work was supported in part by the National Natural Science Foundation of China under Grants 61962004 and 61462005, and this experimental work is supported by the High Performance Computing Center of Guangxi University. (Corresponding author: Cheng Zhong.)

Digital Object Identifier no. 10.1109/TCBB.2022.3172340

With the fast development of biological technology, the chemical biology data in the public databases, such as PubChem[12], ChEMBL[13], KEGG[14], and STITCH[15], have increase to millions over the past 20 decades. The advantage of deep learning method is more obvious in dealing with large scale compound protein interaction pairs. In recent years, many deep learning frameworks have been utilized in drug discovery research [16], [17]. Compared with traditional machine learning methods, deep learning methods have the advantage of extracting high-order feature interactions and mining deep hidden relationship between compounds and proteins.

The method DL-CPI [18] uses PubChem fingerprints of compound molecular and PFam descriptors of protein as input feature vectors, and then trains the prediction model with Deep Neural Networks (DNNs). Regarding compounds and proteins as 1D sequences or word-based sequences, the method DeepDTA [19] uses convolutional neural networks (CNNs) to extract real-valued features of compounds and proteins. The method WideDTA [20] adapts the word-based sequence representation for compounds and proteins, and utilizes two extra features LMCS (ligand max common structures) and PDM (protein motifs and domains) to improve model performance and prediction accuracy. From the perspective that compound structure is regarded as molecular graph, the methods CPI-GNN[21] and GraphDTA[22] use graph neural networks (GNNs) [23], [24] and graph convolutional neural networks (GCNs) [25] to learn representation of compounds, the model GANDTI integrates a graph convolutional autoencoder and generative adversarial network (GAN) to deeply learn the feature vectors for drugs and targets [26]. Regarding both compounds and proteins as sequence data, recurrent neural networks (RNNs) are used to extract feature vectors of compounds and proteins in DeepAffinity [27] and Zheng's work [28]. In addition, attention mechanism is introduced to improve the prediction accuracy, the model TransformerCPI addresses sequence-based CPI classification task by modifying transformer architecture with self-attention mechanism [29], the method MHSADTI predicts DTIs based on the graph attention network and multi-head self-attention mechanism [30].

Deep learning model can achieve good high-order feature interactions of compound molecules and target proteins. However, since the compound-protein interactions are high-order sparse, deep learning model will over-generalize and produce prediction of less relevant drugs when it extracts only the high-order feature interactions. By introducing hybrid architectures with learning both low and high-order feature interactions, the methods Wide&Deep [31] and DeepFM [32] overcome the problem of prediction error caused by data sparsity. The low-order feature interactions can use cross-product transformations over sparse features. However, the method DeepFM [32] uses DNN as the deep part, which is suitable for learning the categorical features in the prediction of click-through rate (CTR), but not suitable for learning compound subgraph features in the CPIs prediction problem.

For the compound-protein interactions, 1-order feature interactions can be obtained directly from the raw feature, e.g., the feature "GetAtomic == O" has value of 1 if the compound contains oxygen atoms. 2-order feature interactions can be achieved effectively by using cross-product transformations over sparse features. For example, AND (GetAtomic == S,

GetFormalCharge == 0) has value of 1 if the compound contains sulfur atoms and the sulfur atoms have no charge. The 1-order and the 2-order feature interactions are defined as the low-order feature interactions, and the combination of the 3-order and over 3-order feature interactions is defined as the high-order feature interactions. The low-order and high-order feature interactions correlate with the final compound-protein interaction.

Inspired by the model DeepFM [32], we proposed a new hybrid model called FMGNN to learn both low and high-order feature interactions. Learning the low-order feature interactions can find the frequent co-occurrence of features. Learning the high-order feature interactions can explore implicit feature interactions. The model FMGNN integrates the architectures of factorization machine (FM) [33] and graph neural network (GNN) [23] to learn the low and high-order feature interactions of compound graph, and integrates the architectures of FM and convolutional neural network (CNN) to learn the low and high-order feature interactions of protein sequences. The feature interactions of compounds and protein sequences are concatenated to predict CPIs.

Our main contributions are summarized as follows:

- 1) We propose a novel model called FMGNN that integrates the architectures of factorization machine (FM) and graph neural network (GNN). The FMGNN builds prediction model using low-order feature interactions of compounds and proteins with FM and it also builds prediction model using high-order feature interactions of compounds and proteins with GNN and CNN. The model FMGNN can learn low and high-order feature interactions concurrently.
- 2) We generate the compound feature vectors with compound substructure graphs and pharmacophore features, which consider not only the topological similarity, but also the functional similarity between compound subgraphs. This ensures that the prediction results much more fit the expectation of biological experiment.
- 3) We construct the gram corpus and treat it as the pre-trained model in the embedding layer of CNN model, which can reduce the training iteration times and improve the convergence rate of the proposed model.

2 MATERIAL AND METHODS

2.1 Material

The experimental data used are the datasets for human and *C. elegans* that created by Liu *et al.* [34]. They include highly credible negative samples of compound-protein pairs obtained by a systematic screening framework. The positive samples were retrieved from DrugBank [35] and Matador [36]. The dataset human contains 3364 positive interactions between 1052 compounds and 852 proteins. The dataset *C. elegans* contains 4000 positive interactions between 1434 compounds and 2504 proteins.

To inspect our proposed prediction method on large-scale data, we retrieved the compound-protein pairs of *Homo sapiens* from database STITCH Version 5.0 [15]. To ensure the highly credible samples, we retrieved the compound-protein pairs that their interaction probability is greater than 90% as positive samples, and lower than 10% as negative samples.

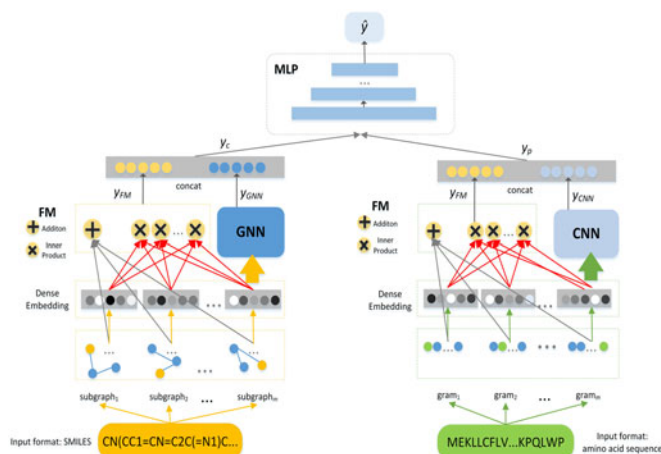


Fig. 1. Framework of FMGNN.

The final dataset STITCH used contains 115927 positive interactions between 13286 compounds and 5313 proteins.

Because real CPI datasets are typically imbalanced, we evaluate the robustness of the prediction methods by the imbalanced dataset. Fixed the number of positive samples, we set three ratios of positive and negative samples as 1:1, 1:3, and 1:5 respectively in our experiment. This kind of experimental setting was first proposed by Tabei and Yamashita [37]. All the negative samples were retrieved from the low candidates based on the scores obtained by the database STITCH. As a classification problem, the metrics such as the AUC, precision, recall and F1-score are used to evaluate CPI prediction performance.

2.2 Method

In this paper, we propose a novel prediction model called FMGNN, which learn both low and high order feature interactions, to predict compound-protein interactions. The model FMGNN integrates the architectures of factorization machine (FM) [33] with graph neural network (GNN) [21], [23] and graph convolutional neural networks (GCNs) [21], [25], respectively. Fig. 1 shows the framework of FMGNN.

As illustrated in Fig. 1, the compound with SMILES notation and protein amino acid sequence are two inputs of the prediction model FMGNN. Compounds are represented by a molecule graph with atoms as nodes and chemical bonds as edges (details in Section 2.2.1), and proteins are represented as a word sequence with fixed amino acids sub-sequences as words. The model FMGNN jointly trains with FM and GNN to learn low and high-order feature interactions among substructure graphs of compounds, and outputs the combined feature vector of compound $y_C = f(y_{FM} + y_{GNN})$, and then the model FMGNN jointly trains with FM and CNN to learn low and high order feature interactions among sub-sequences of amino acids, and outputs the combined feature vector of protein $y_P = f(y_{FM} + y_{CNN})$. Finally, the model FMGNN concatenates two feature vectors and passes through fully-connected layers and a softmax layer to calculate the final output \hat{y} :

$$\hat{y} = \text{Softmax}(\text{ReLU}(y_C + y_P)) \quad (1)$$

where $\hat{y} \in (0,1)$ is the predicted CPI probability, y_C is the feature vector of compounds, y_P is the feature vector of proteins, and ReLU is a non-linear activation function [38].

2.2.1 Compound Substructure Graphs With Pharmacophore Features

In this section, we introduce compound substructure graphs with pharmacophore features. We use r -radius subgraphs [39] to represent the compound substructures. The r -radius subgraphs are induced by the neighboring vertices and edges within radius r from a vertex. However, the prediction bias may be caused if only the substructure similarity between compounds is considered. The prediction bias will be corrected if the pharmacophore features of compound molecules, such as hydrogen bond acceptors and hydrogen bond donors, are taken into account. So we use the pharmacophore features of compound molecules in the process of constructing the substructure graphs of compound molecules. In our work, we used 7 types of pharmacophore features, including hydrogen bond donors, hydrogen bond acceptors, aromatic, posIonizable, negIonizable, hydrophobe and ZnBinder, to construct the substructure graphs of compound molecules.

Pharmacophore [40] is used to featurize the compound molecules by identifying essential properties of molecular recognition. Every type of atom or group in a compound can be reduced to a pharmacophore feature, which can be used to analyze the similarity among small molecules and identify the key contributing features to the biological function. In pharmacophore-based model, the concept of bioisosterism is used to guarantee the model more reliable, which considers not only the topological similarity of molecules, but also the functional similarity of groups.

For example, Fig. 2 shows the topological structure-based and pharmacophore-based alignments between methotrexate and dihydrofolate, respectively. By comparing with the conformation superposition (1rx2, 1rb3) verified by experiment, we can see that the pharmacophore based conformation is closer to the experimental result. Therefore, we will obtain higher prediction accuracy by adding the pharmacophore features of molecules than using only the topological structure of compounds in learning representation of r -radius substructure graphs of compound.

To describe the compound substructure graphs with pharmacophore features, we use a graph $\mathcal{G} = (\mathcal{V}(\mathcal{A}, \mathcal{P}), \mathcal{E})$, where $\mathcal{V}(\mathcal{A}, \mathcal{P})$ is the set of atoms, \mathcal{A} is the set of atom types, \mathcal{P} is the set of pharmacophore features of \mathcal{A} , \mathcal{E} is the set of chemical bonds between adjacent atoms, and $e_{ij} \in \mathcal{E}$ is the chemical bond connecting the i -th and j -th atoms. For atom v_i , $v_i(a_i, p_i) \in \mathcal{V}(\mathcal{A}, \mathcal{P})$ represents the i -th atom with atom type a_i and pharmacophore feature p_i . Firstly, we embed all atoms and chemical bonds in a d -dimensional real-valued vector space with these atom types and pharmacophore features. Then, we construct r -radius substructure graphs [39] by the neighboring vertices and edges within radius r from a vertex [21].

We defined a set $\mathcal{N}(i, r)$ to represent neighboring atoms within radius r from the i -th atom. Note that $\mathcal{N}(i, 0) = \{i\}$. We define the r -radius substructure graphs for vertex v_i , $v_i^{(r)}$, as follows:

$$v_i^{(r)} = \left(\mathcal{V}_i^{(r)}(a_i, p_i), \mathcal{E}_i^{(r)} \right) \quad (2)$$

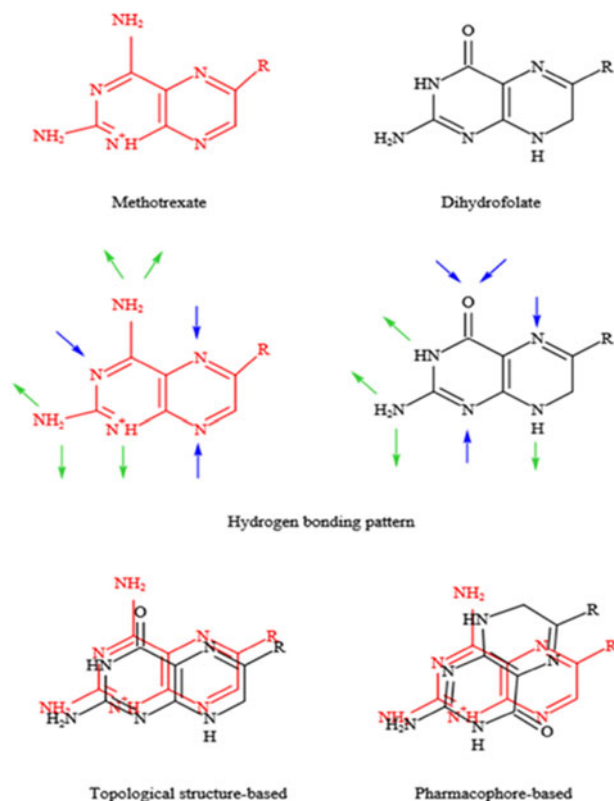


Fig. 2. Comparison between topological structure and pharmacophore.

where

$$\mathcal{V}_i^{(r)}(a_i, p_i) = \{v_j(a_j, p_j) | j \in \mathcal{N}(i, r)\},$$

$$\mathcal{E}_i^{(r)} = \{e_{mn} \in \mathcal{E} | (m, n) \in \mathcal{N}(i, r) \times \mathcal{N}(i, r-1)\}.$$

Then, we define the r -radius substructure graphs for edge e_{ij} , $e_{ij}^{(r)}$, as follows:

$$e_{ij}^{(r)} = \left(\mathcal{V}_i^{(r)}(a_i, p_i) \cup \mathcal{V}_i^{(r-1)}(a_i, p_i), \mathcal{E}_i^{(r)} \cap \mathcal{E}_j^{(r)} \right) \quad (3)$$

Next, the model FM is used to train the substructure graphs of a compound for getting the low-order feature interactions (see: Section 2.2.2). Meanwhile, the model GNN is used to train the substructure graphs of a compound for getting the high-order feature interactions (see: Section 2.2.3).

2.2.2 Factorization Machines (FM) Model

The model FM [33] is a factorization machine to learn feature interactions for recommendation system. The model FM combines the advantages of Support Vector Machines (SVM) with factorization models to estimate feature interactions using factorized parameters reasonably in very sparse data. The computation complexity of the model FM is linear, and its optimization effect is good. The model FM is a general predictor working with any real value feature vector. So we choose the model FM to learn low-order feature interactions in CPIs prediction. The procedure to learn compound feature vectors with model FM is shown in Fig. 3.

As shown in Fig. 3, the substructure graphs of a compound (Section 2.2.1) are the input of the model FM. When both substructure graph i and substructure graph j appear in the same compound, the output of component FM, y_{FM} , is the

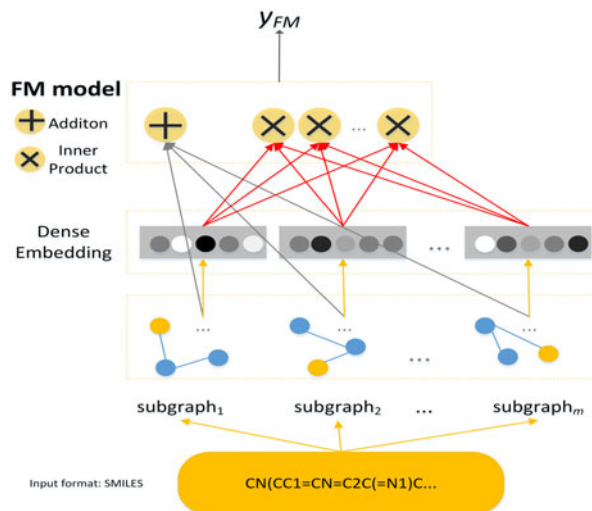


Fig. 3. Procedure to learn compound feature vectors with model FM.

summation of weighted 1-order and 2-order feature interactions:

$$y_{FM} = w_0 + \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j=i+1}^n w_{i,j} \cdot x_i \cdot x_j \quad (4)$$

where x_i , x_j are the i -th and j -th substructure graphs of a compound respectively, w_0 is the global bias, w_i is the parameter of x_i , $w_{i,j}$ is the parameter of the interaction between x_i and x_j , $i, j = 1, 2, \dots, n$.

For any positive definite matrix W , there exists a matrix V such that $W = V \times V^T$, where $V \in \mathbb{R}^{n \times k}$ and V^T is the transposition of V .

The 2-order interaction between x_i and x_j can be learned via the inner product of their latent vectors v_i and v_j .

$$\sum_{i=1}^n \sum_{j=i+1}^n w_{i,j} \cdot x_i \cdot x_j = \sum_{i=1}^n \sum_{j=i+1}^n \langle v_i, v_j \rangle x_i \cdot x_j \quad (5)$$

where $\langle v_i, v_j \rangle$ is the dot product of two latent vectors v_i and v_j with size k :

$$\langle v_i, v_j \rangle = \sum_{f=1}^k v_{i,f} \cdot v_{j,f} \quad (6)$$

Then, the 2-order interaction between x_i and x_j can be reformulated as follow [33]:

$$\sum_{i=1}^n \sum_{j=i+1}^n \langle v_i, v_j \rangle x_i x_j = \frac{1}{2} \sum_{f=1}^k \left(\left(\sum_{i=1}^n v_{i,f} x_i \right)^2 - \sum_{i=1}^n v_{i,f}^2 x_i^2 \right) \quad (7)$$

Meanwhile, the output of FM component, y_{FM} , is rewritten as follow:

$$y_{FM} = w_0 + \sum_{i=1}^n w_i x_i + \frac{1}{2} \sum_{f=1}^k \left(\sum_{i=1}^n v_{i,f} x_i \right)^2 - \sum_{i=1}^n v_{i,f}^2 x_i^2 \quad (8)$$

While the FM can model high-order feature interactions, in practice only 2-order feature interactions are usually

considered due to high complexity. The models GNN [23] and CNN [21] are applied to learn high-order feature interactions for compound molecule graphs and protein amino acids sequences, respectively.

2.2.3 Model GNN

In our work, the model GNN is based on learning representations of r -radius substructure graphs of compound. The model GNN maps a graph \mathcal{G} to a vector $y \in \mathbb{R}^d$ with two functions, *i.e.*, transition and output functions [21], [23].

In the model GNN, firstly, the feature vector for the substructure graphs of a compound is randomly initialized in embedding layer, the i -th substructure graph embedding at time step t is represented as $v_i^{(t)}$, and $v_i^{(t)}$ is updated with transition function [21]:

$$v_i^{(t+1)} = \text{sigmoid}\left(v_i^{(t)} + \sum_{j \in \mathcal{N}(i)} h_{ij}^{(t)}\right) \quad (9)$$

where sigmoid is the activation function, $\mathcal{N}(i)$ is the set of neighboring indices of the i -th substructure graph, and $h_{ij}^{(t)} \in \mathbb{R}^d$ is the hidden neighborhood vector.

Then, the model GNN learns the neural network parameters including the feature vectors via back propagation to obtain the final output y_{GNN} :

$$y_{GNN} = \frac{1}{|\mathcal{V}|} \sum_{i=1}^{|\mathcal{V}|} v_i^{(t)} \quad (10)$$

where $|\mathcal{V}|$ is the number of substructure graphs of a compound.

Random initialization operation will spend a long time to achieve convergence in the GNN training. To accelerate the training for protein feature vector, we construct a gram corpus with the physicochemical properties of amino acids for proteins.

2.2.4 CNN Model With Physicochemical Properties of Amino Acids

As shown in Fig. 4, the models FM [33] and CNN [21] are used to embed the n -gram amino acids into vector, and then obtain low and high-order feature vectors for protein sequences, respectively. The model FM can obtain the low-order feature interactions of protein sequences as described in Section 2.2.2. In this section, we describe the model CNN, which can obtain high-dimensional real-valued vector representations of protein sequences. The model CNN maps sequence $S = \{s_1^0, s_2^0, \dots, s_{|S|}^0\}$ to vector $y_{CNN} \in \mathbb{R}^d$ with multiple filter functions in t times, $s_i^{(t)} = \text{ReLU}(W_{conv}v_i^{(t-1)} + b_{conv})y_{CNN} = \frac{1}{|S|} \sum_{i=1}^{|S|} s_i^{(t)}$ where the dimensionality d of protein sequences is the same as that of the compound substructure graphs described in Section 2.2.1. Fig. 4 shows the procedure to learn protein feature vectors with the models FM and CNN.

To apply the model CNN to deal with protein sequence, the protein sequences is first divided into overlapping n -gram amino acids, and then the n -gram amino acids are defined as “words” [41]. Since a protein consists of 20 amino acids, the number of all possible n -grams is 20^n . To keep the vocabulary of reasonable size and avoid low-frequency

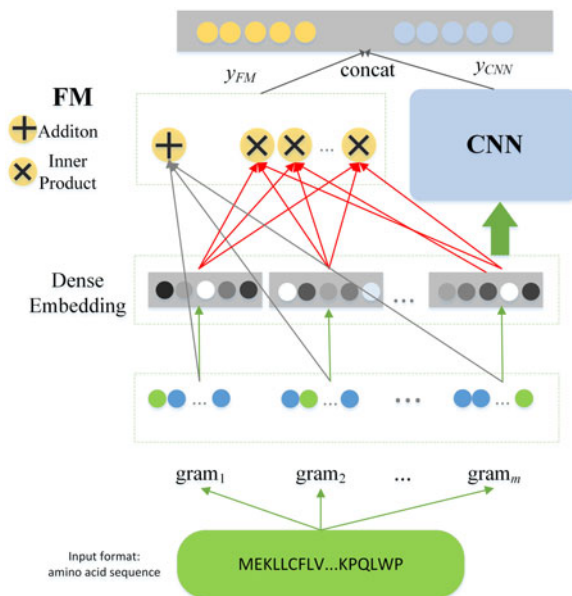


Fig. 4. Procedure to learn protein feature vectors with models FM and CNN.

words in the learning representations, we set an n -gram number $n = 3$. For example, we divide an adenosine deaminase-like protein into an overlapping 3-gram amino acid sequence as follow:

MAQTP...GQNL \rightarrow “MAQ”, “AQT”, “QTP”, ..., “GQN”, “QNL”.

To accelerate training the model CNN, we construct a gram corpus for “words” based on 554 physicochemical properties of amino acids. These physicochemical properties of amino acids are obtained from the dataset AAindex1 [42].

The dataset AAindex1 adopts different dimensions for different attributes of amino acids, and the differences between their values are also very large, the direct use of the dataset AAindex1 will affect the results of data analysis. We use Z normalization method to normalize the original data from the dataset AAindex1. A normalization term f is defined as follows:

$$f = \frac{f_o - \bar{f}}{\sigma} \quad (11)$$

where f_o is the original data, \bar{f} is the mean value of f_o , σ is the standard deviation of f_o .

Given a set of amino acids $AA = \{x_1, x_2, \dots, x_{20}\}$, where each amino acid $x_i = \{f_{i,1}, f_{i,2}, \dots, f_{i,554} \mid i \in [1], [20]\}$ is a vector with the Z normalized 554 physicochemical properties, we construct gram corpus $C = \{gram_1, gram_2, \dots, gram_{|gram|}\}$, where $|gram|$ is the total number of grams, $|gram| = 20^n$. For $n = 3$, each 3-gram $gram(x_i, x_j, x_k) = \frac{1}{n}(x_i + x_j + x_k)$ is a feature vector, where $x_i, x_j, x_k \in \{x_1, x_2, \dots, x_{20}\}$ are any three amino acids, the addition operation of vector is the addition of corresponding components of vector. Hence, we have

$$\begin{aligned} & \frac{1}{n}(x_i + x_j + x_k) \\ &= \left\{ \frac{f_{i,1} + f_{j,1} + f_{k,1}}{n}, \frac{f_{i,2} + f_{j,2} + f_{k,2}}{n}, \dots, \frac{f_{i,554} + f_{j,554} + f_{k,554}}{n} \right\} \end{aligned} \quad (12)$$

Next, the principal component analysis (PCA) was applied to keep only the most important features by

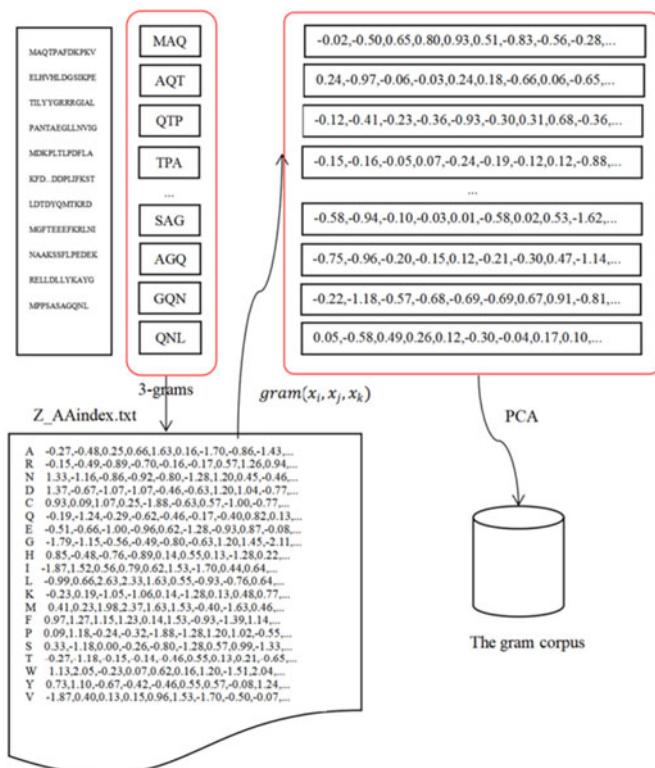


Fig. 5. Constructing process of 3-gram vectors for an adenosine deaminase-like protein.

removing the noise and unimportant features. In our study, we set $N_{PCA} = d$, d is the same as the dimensionality of protein sequences. Finally, the gram corpus C is loaded as the pretrained model in the embedding layer of model CNN.

For example, the construction of gram corpus for an adenosine deaminase-like protein is shown in Fig. 5. The Z_AAindex.txt file stores the Z normalized 554 physicochemical properties from the dataset AAindex1.

2.2.5 Algorithm

Based on the above steps, we propose an algorithm for predicting compound-protein interactions integrating the architectures of models FM and GNN called FMGNN, in which its input file "CPIFile" includes the positive and negative CPI pairs extracted from database, the file "Z_AAindex.txt" stores the normalized 554 physicochemical properties from the dataset AAindex1, r is the radius of substructure graphs and the default value of r is 2, and n is the number of amino acids in a gram and its default value of $n = 3$. Algorithm 1 describes our proposed prediction algorithm called FMGNN.

Our algorithm FMGNN can obtain more accurate prediction than other existing algorithms because FMGNN learns high-order feature interactions to mine deep hidden relationship between compounds and proteins, extracts low-order feature interactions to solve the problem of over-generalized and produced prediction of less relevant drugs when the compound-protein interactions are high-order sparse; and it uses pharmacophore features of compound to ensure that the prediction results much more fit the expectation of biological experiment,

and refers the physicochemical properties of amino acids to improve the convergence speed and accuracy of protein feature extraction.

Algorithm 1 FMGNN

Input: CPIFile, Z_AAindex.txt, r , n

Output: AUC, Precision, Recall, F1-score

Begin

1. Construct r -radius substructure graphs with pharmacophore features for compounds. For i -th vertex, $v_i^{(r)} = (\mathcal{V}_i^{(r)}(a_i, p_i), \mathcal{E}_i^{(r)})$, for edge e_{ij} between the i -th and j -th vertex, $e_{ij}^{(r)} = (\mathcal{V}_i^{(r)}(a_i, p_i) \cup \mathcal{V}_i^{(r-1)}(a_i, p_i), \mathcal{E}_i^{(r)} \cap \mathcal{E}_j^{(r)})$, as described in Section 2.2.1;
2. Divide the protein sequences into overlapping n -gram amino acids;
3. Construct the gram corpus $C = \{c_1, c_2, \dots, c_{|C|}\}$ for all the n -gram amino acids;
4. For i -th vertex, initialize the feature vector of compound x_i randomly in the embedding layer.
5. Compute the low-order feature vector of compound y_{FM-c} with the model FM as described in Section 2.2.2, where $y_{FM-c} = w_0 + \sum_{i=1}^n w_i x_i + \frac{1}{2} \sum_{f=1}^k ((\sum_{i=1}^n v_{i,f} x_i)^2 - \sum_{i=1}^n v_{i,f}^2 x_i^2)$;
6. Compute the high-order feature vector of compound y_{GNN} with model GNN, $y_{GNN} = \frac{1}{|V|} \sum_{i=1}^{|V|} v_i^{(t)}$, as described in Section 2.2.3;
7. Concatenate the low and high-order feature vectors of compound to generate the final compound feature vector y_C , $y_C = concatenate(y_{FM-c}, y_{GNN})$;
8. For the i -th gram, initialize the feature vector of protein g_i with the gram corpus C as the pretrained weight in the embedding layer.
9. Compute the low-order feature vector of protein sequence y_{FM-p} with the model FM as described in Section 2.2.2, where $y_{FM-p} = w_0 + \sum_{i=1}^n w_i g_i + \frac{1}{2} \sum_{f=1}^k ((\sum_{i=1}^n v_{i,f} g_i)^2 - \sum_{i=1}^n v_{i,f}^2 g_i^2)$;
10. Compute the high-order feature vector of protein sequence y_{CNN} with the model CNN, $y_{CNN} = \frac{1}{|g|} \sum_{i=1}^{|g|} g_i^{(t)}$, described in Section 2.2.4;
11. Concatenate the low and high-order feature vectors of protein sequence to generate the final protein feature vector y_P , $y_P = concatenate(y_{FM-p}, y_{CNN})$;
12. Concatenate y_C and y_P , and pass through fully-connected layers to calculate the final output $\hat{y} = ReLU(W(y_C + y_P) + b)$.

End.

3 EXPERIMENT

For the compounds, we taken the SMILES notation of the compounds as input, which was converted to a graph representation, and extracted information of the molecular graph with tool RDKit, such as atom types, pharmacophore features, chemical bonds, and the adjacency list of atoms. For proteins, we taken amino acid sequences as input. We normalized 554 physicochemical properties of amino acids from the dataset AAindex1, and constructed an n -gram corpus to accelerate the embedding process for protein sequences.

We implemented our proposed algorithm FMGNN by using Pytorch 1.4.0 with CUDA 10.0 and RDKit 2020.03.3.

TABLE 1
Hyper-parameters in Algorithm FMGNN

Name	value
Number of GNN layers	3
Number of CNN layers	3
Number of output layers	3
Dimension of feature vectors	10
Learning rate	1e-4
Learning rate decay	0.01
Decay interval	10
Weight decay	1e-4
Iteration time	100
Optimizer	LookAhead+RAdam

TABLE 2
Experimental Results of Nine Algorithms on Dataset Human

Algorithm	AUC	Precision	Recall	F1-score
KNN	0.860	0.927	0.798	0.858
RF	0.940	0.897	0.861	0.879
SVM	0.910	0.964	0.969	0.968
GraphDTA	0.960	0.882	0.912	0.897
GCN	0.956	0.862	0.928	0.894
CPI-GNN	0.970	0.918	0.923	0.921
TransformerCPI	0.973	0.916	0.925	0.921
DeepFM	0.525	0.636	0.263	0.372
FMGNN	0.984	0.976	0.953	0.959

We used the optimizers LookAhead [43] and RAdam [44] to train our proposed prediction model. The use of combining LookAhead with RAdam can solve the serious convergence problem caused by the optimizer Adam without learning rate warmup. The experiment was conducted at the CPU/GPU server at high-performance computing center of Guangxi University¹. The experimental configuration is shown at <https://hpc.gxu.edu.cn/gk1/yjzy.htm>. All the settings and hyper-parameters in algorithm FMGNN are summarized in Table 1.

To compare with the traditional machine learning methods, we chose K nearest neighbors (KNN), random forest (RF), and support vector machines (SVM), whose results are obtained by Liu *et al.* [34]. To compare with other deep learning methods, we chose four methods CPI-GNN[21], GraphDTA[22], GCN[25], and TransformerCPI [29]. To compare with relative method, we also chose the method DeepFM [32]. The performance of our algorithm FMGNN with the above eight algorithms was compared in terms of *AUC*, *Precision*, *Recall* and *F1-score*. Tables 2 and 3 show the experimental results for nine algorithms on datasets human and C.elegans, respectively.

From Tables 2 and 3, we can see that compared with other eight algorithms, our proposed algorithm FMGNN achieved higher values of *AUC* and *precision* on both two datasets human and C.elegans. The algorithm SVM achieved higher values of *recall* and *F1-score* than the algorithm FMGNN on dataset human, and the algorithm TransformerCPI achieved higher values of *recall* and *F1-score* than the algorithm

TABLE 3
Experimental Results of Nine Algorithms on Dataset C.Elegans

Algorithm	AUC	Precision	Recall	F1-score
KNN	0.858	0.801	0.827	0.814
RF	0.902	0.821	0.844	0.832
SVM	0.894	0.785	0.818	0.801
GraphDTA	0.974	0.927	0.912	0.919
GCN	0.975	0.921	0.927	0.924
CPI-GNN	0.978	0.938	0.929	0.933
TransformerCPI	0.988	0.952	0.953	0.952
DeepFM	0.535	0.523	0.847	0.647
FMGNN	0.990	0.969	0.931	0.949

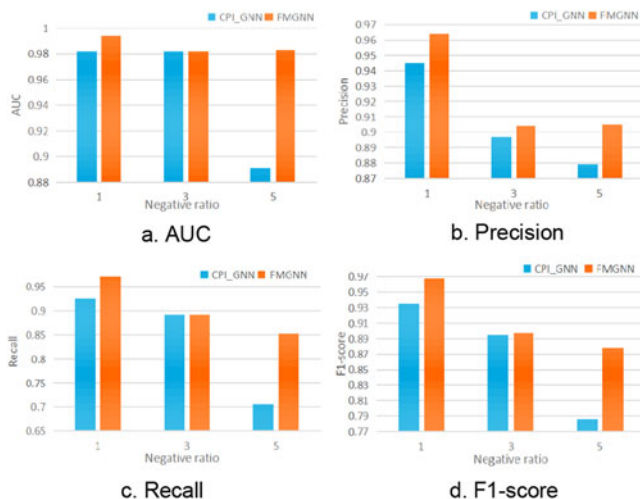


Fig. 6. Experimental results in different negative ratio for the algorithms FMGNN and CPI_GNN on dataset STITCH.

FMGNN on dataset C.elegans. On the whole, the experimental results of *AUC*, *precision*, *recall* and *F1-score* show that our proposed algorithm FMGNN has advantages.

To inspect our proposed algorithm FMGNN on the balanced and imbalanced large-scale datasets, we conducted the experiment on the dataset STITCH, where the dataset STITCH is a larger and much sparser dataset than datasets human and C.elegans. Both the two algorithms FMGNN and CPI_GNN are GNN-based prediction algorithms. We conducted the experiment for algorithms FMGNN and CPI_GNN on the dataset STITCH. The experimental results are shown in Fig. 6.

As shown in Fig. 6, compared to the algorithm CPI_GNN, our proposed algorithm FMGNN obtained higher values for *AUC*, *Precision*, *Recall* and *F1-score*. It illustrates that our algorithm FMGNN is robust even if the dataset is imbalanced and larger one. Moreover, the more imbalanced the data is, the higher the performance of our algorithm is. i.e., the *AUC* score increased 9% when the negative ratio is 5, and only 1% when the negative ratio is 1; the *Recall* score increased 15% when the negative ratio is 5, and only 5% when the negative ratio is 1; the *F1-score* increased 9% when the negative ratio is 5, and only 3% when the negative ratio is 1. It indicates that the more spares the dataset is, the more efficient of considering both low and higher-order features is, and the more important the low-order feature interactions is.

1. <http://hpc.gxu.edu.cn>

TABLE 4
Experimental Results for FMGNN and FMGNN Without Layer FM on Dataset STITCH

Algorithm	AUC	Precision	Recall	F1-score
FMGNN w/o FM	0.970	0.923	0.918	0.920
FMGNN w/ FM	0.983	0.963	0.936	0.949

4 DISCUSSION

4.1 Effect of Learning Both Low and Higher-Order Feature Interactions

Some prediction methods achieve high performance in CPI with the learning models CNN, DNN, and GNN [18], [19], [20], [21], [22]. The major contribution of these deep learning models is that they explore new implicit feature interactions. But along with the increase of compounds, the compound-protein pairs become high-order sparse like the pairs in dataset STITCH, the models CNN, DNN, and GNN [18], [19], [20], [21], [22] will over-generalize and produce prediction of less relevant drugs. The major downside of these deep learning models is that they focus more on high-order feature interactions while ignore low-order feature interactions. In general, there are sophisticated feature interactions between compounds and proteins in CPI prediction, learning both low and high-order feature interactions can find the frequent co-occurrence of features, and explore implicit feature interactions. We integrated model FM with models GNN and CNN in our proposed prediction model. This allows our proposed model to improve the prediction accuracy of CPI by jointly learning low and high-order feature interactions.

Learning both low and high-order features brings additional improvement over the case of learning only one feature. To show the effect of learning both low and high-order feature interactions, we conducted experiment on dataset STITCH for our algorithm FMGNN with and without the layer FM, respectively. The experimental results are shown in Table 4.

From Table 4 we can see that compared to the algorithm FMGNN without layer FM, the algorithm FMGNN with FM obtained higher values for all the metrics on dataset STITCH. It indicates that low-order feature interactions contribute to the prediction results.

4.2 Effect of Pharmacophore Features

Pharmacophore [40] has great significance in process of drug discovery. There are seven kinds of pharmacophore features, including hydrogen bond donor, hydrogen bond acceptor, positive and negative charge center, aromatic ring center, hydrophobic group, hydrophilic group, and geometric conformation volume collision. The methods based on pharmacophore features make use of not only the topological similarity of compounds but also the functional similarity of groups. Thus, using the concept of bioisosterism makes the prediction model more reliable.

To show the effect of pharmacophore features, we conducted the experiment for the algorithms FMGNN and FMGNN without pharmacophore features on dataset STITCH. The experiment results are shown in Table 5.

We can see from Table 5 that compared to the algorithm FMGNN without pharmacophore features, the algorithm FMGNN with pharmacophore features obtained higher

TABLE 5
Experimental Results for FMGNN and FMGNN Without Pharmacophore Features on Dataset STITCH

Algorithm	AUC	Precision	Recall	F1-score
FMGNN w/o pharmacophore features	0.970	0.923	0.918	0.920
FMGNN w/ pharmacophore features	0.984	0.977	0.944	0.960

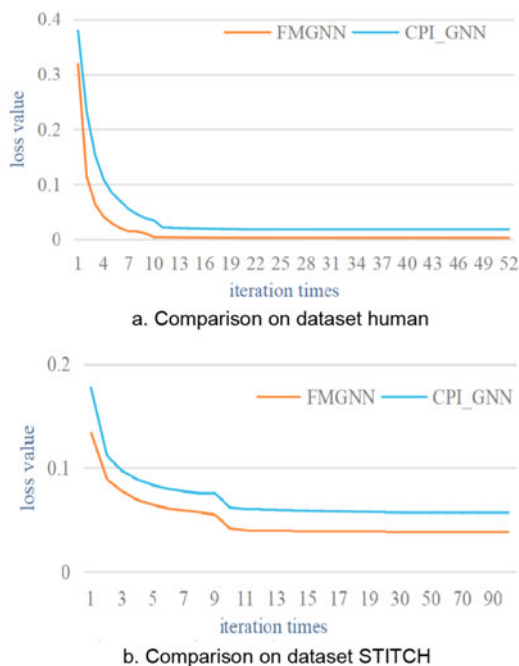


Fig. 7. Convergence comparison for the algorithms FMGNN and CPI_GNN.

values for all the metrics on dataset STITCH. It means that pharmacophore features are contributed to the prediction results.

4.3 Convergence Effect of Gram Corpus

In general, randomly initialization operation is used to embed feature vector in the embedding layer for deep learning model. But random initialization makes the algorithm take a long time to achieve convergence and have a large loss value for large-scale dataset. To accelerate the algorithm converge and reduce the loss value for the predicted value and real value, we used a gram corpus to initialize the weight of embedding layer. The gram corpus was constructed by the physicochemical properties of amino acids for proteins based on the dataset AAindex1, and it was loaded as the pretrained model for embedding layer in the model CNN. To evaluate the convergence effect of gram corpus, we conduct the experiment for the algorithms FMGNN and CPI_GNN on datasets STITCH and human. Fig. 7 shows the experiment result.

As shown in Fig. 7, our proposed algorithm FMGNN spent a short time to achieve convergence and had a less loss value on both two datasets. The loss value was stable at 40 iterations with the algorithm FMGNN, but stable at 50



Fig. 8. Comparison on training loss value, elapsed time and four metric for linear GNN, GCN and GATs layer as the deep part in algorithm FMGNN on dataset Human.

iterations with the algorithm CPI_GNN on both two datasets. The details are given in Table S1-Table S4.

For the human dataset, the average loss value was 0.0025 in our proposed algorithm, but 0.0178 in the algorithm CPI_GNN. For the dataset STITCH, the average loss value is 0.0384 in our algorithm FMGNN, but 0.0569 in algorithm CPI_GNN. The details of average loss values are given in Table S5. This indicates that our proposed algorithm FMGNN had a less loss value, and the gram corpus contributed to the low computational cost.

4.4 Comparing With Different GNNs

We used the linear GNN [23] as the deep part to learn high-order features for compounds in our proposed algorithm. As we know, there are other popular GNN models, such as GCN [25] and GATs [45]. To evaluate the contribution of

TABLE 6
Top 5 Candidate Target Proteins for Wogonin

UniprotID	Protein Name	Score
P36542	ATP5F1C	0.972
P62424	RPL7A	0.970
O75822	EIF3J	0.968
O00425	IGF2BP3	0.968
P18754	RCC1	0.968

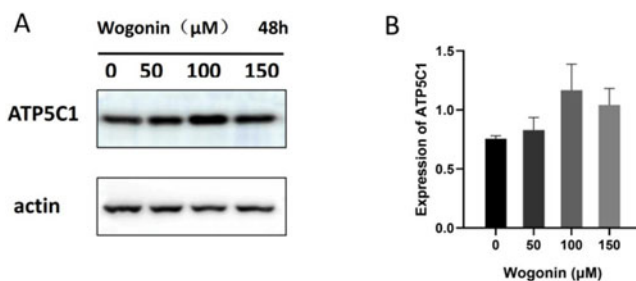


Fig. 9. (A) Hep G2 cells were treated with 0 μM , 50 μM , 100 μM and 150 μM of wogonin for 48 hours, and then the expression levels of protein ATP5C1 were detected by western blot ($n = 3$). (B) Quantification of protein ATP5C1 obtained by using Image J software. The values were the ratios of ATP5C1/actin ($n = 3$, ns, not significant, * $P < 0.05$).

various GNN models in our algorithm FMGNN, we conducted the experiment for linear GNN, GCN and GATs as the deep part layer to extract high-order feature interactions for compounds on the dataset Human. Fig. 8 shows the experiment result.

As shown in Fig. 8, the GATs layer spent a shortest time to finish the training. The GCN layer had the minimum training loss value. And the GNN layer had the best metric values. In practical applications, we should choose the linear GNN or GCN or GATs according to the actual dataset and environment.

4.5 Case Study

To verify the reliability of the algorithm FMGNN, we predicted the interactions between wogonin and 300 candidate target proteins. Wogonin is a dihydroxy-and monomethoxy-flavone. It has a role as a cyclooxygenase 2 inhibitor, an antineoplastic agent, an angiogenesis inhibitor, and a plant metabolite. Its SMILES expression is "COC1 = C (C2 = c1oc (= CC2 = O) C3 = CC = CC = C3) O". We choose the algorithm FMGNN trained by dataset STITCH as prediction model. After calculating prediction scores by algorithm FMGNN, the prediction scores of 300 candidate target proteins were ranked, and the top 5 proteins were shown in Table 6. Furthermore, the Top 1 protein was selected for biological experiment verification, and the experimental results were shown in Fig. 9.

The ATP5C1 encodes a subunit of mitochondrial ATP synthase, which catalyzes the synthesis of ATP utilizing an electrochemical gradient of protons across the inner membrane during oxidative phosphorylation [46]. It has been reported that the expression of protein ATP5C1 in liver tumor tissues was lower than that in non-tumor tissues [47]. Our western blot experiment results showed that compared with the control group, the expression of protein ATP5C1 was increased as HepG2 cells were exposed to 50 μM ,

100 μ M and 150 μ M of wogonin for 48 hours. This indicates that protein ATP5C1 may be a potential target of wogonin.

5 CONCLUSION AND FUTURE WORK

In this paper, we propose a hybrid model of incorporating Factorization Machines (FM) and GNN/CNN with pharmacophore features of compounds and physicochemical properties of amino acids to predict potential compound-protein interactions. The model FM was applied to extract the low-order feature interactions, the models GNN and CNN were used to learn the high-order feature interactions for compounds and proteins, respectively. The experimental results demonstrated that a jointly low and high-order feature interactions obtained additional improvement of CPIs prediction. Generating the compound feature vectors with compound substructure graphs and pharmacophore features can ensure the prediction results closer to the expectation of biological experiment. Loading the physicochemical properties of amino acids as the pre-trained weight model in embedding layer can accelerate the training process of the model CNN. The experimental results on datasets Human and C.elegans showed that our proposed prediction method outperformed the classical machine learning methods and existing deep learning methods in term of AUC and precision. In addition, the experimental results on a large-scale balance and imbalance dataset also showed that our proposed algorithm outperformed the algorithm CPI_GNN in all terms of AUC, precision, recall and F1-score. The western blot experiment results on wogonin and its candidate target proteins showed that our algorithm FMGNN was effective and accurate for finding potential target proteins.

The target of drugs usually refers to protein, but RNA is also a potential target. One future research direction is how to predict the drug-RNA interactions because the known drug-RNA interaction data is scarce. Meanwhile, it is also another future research direction for predicting drug-target interactions using deep learning framework on multiomics information such as gene regulatory omics and metabolome.

ACKNOWLEDGMENTS

The authors thank the editor and anonymous reviewers for their constructive comments and suggestions, which greatly help us improve our manuscript.

REFERENCES

- P. Csermely, T. Korcsmáros, H. Kiss, G. London, and R. Nussinov, "Structure and dynamics of molecular networks: A novel paradigm of drug discovery. A comprehensive review," *Pharmacol. Therapeutics*, vol. 138, no. 3, pp. 333–408, 2013.
- D. Hao, T. Ichigaku, M. Hiroshi, and S. Zhu, "Similarity-based machine learning methods for predicting drug-target interactions: A brief review," *Brief. Bioinf.*, vol. 15, no. 5, pp. 734–747, 2014.
- B. Maryam, S. Elyas, K. Wang, M. A. Sartor, N. C. Zaneta, and N. Kayvan, "Machine learning approaches and databases for prediction of drug-target interaction: A survey paper," *Brief. Bioinf.*, vol. 22, no. 1, pp. 247–269, 2020.
- Y. Yamanishi, M. Araki, A. Gutteridge, W. Honda, and M. Kanehisa, "Prediction of drug-target interaction networks from the integration of chemical and genomic spaces," *Bioinformatics*, vol. 24, no. 13, pp. i232–i240, 2008.
- K. Bleakley and Y. Yamanishi, "Supervised prediction of drug-target interactions using bipartite local models," *Bioinformatics*, vol. 25, no. 18, pp. 2397–2403, 2009.
- L. van, N. Twan, B. Sander, and E. Marchiori, "Gaussian interaction profile kernels for predicting drug-target interaction," *Bioinformatics*, vol. 27, no. 21, pp. 3036–3043, 2011.
- M. C. Cobanoglu, C. Liu, F. Hu, Z. N. Oltvai, and I. Bahar, "Predicting drug-target interactions using probabilistic matrix factorization," *J. Chem. Inf. Model.*, vol. 53, no. 12, pp. 3399–3409, 2013.
- Y. Liu, W. Min, C. Miao, P. Zhao, and X. L. Li, "Neighborhood regularized logistic matrix factorization for drug-target interaction prediction," *PLOS Comput. Biol.*, vol. 12, no. 2, 2016, Art. no. e1004760.
- H. Ming, S. H. Bryant, and Y. Wang, "Predicting drug-target interactions by dual-network integrated logistic matrix factorization," *Sci. Rep.*, vol. 7, 2017, Art. no. 40376.
- X. Chen, M. X. Liu, and G. Y. Yan, "Drug-target interaction prediction by random walk on the heterogeneous network," *Mol. Biosyst.*, vol. 8, no. 7, pp. 1970–1978, 2012.
- C. Tang, C. Zhong, D. Chen, and J. Wang, "Drug-target interactions prediction using marginalized denoising model on heterogeneous networks," *BMC Bioinf.*, vol. 21, 2020, Art. no. 330.
- Y. Wang, J. Xiao, T. O. Suzek, J. Zhang, J. Wang, and S. H. Bryant, "PubChem: A public information system for analyzing bioactivities of small molecules," *Nucleic Acids Res.*, vol. 37, no. suppl_2, pp. W623–W633, 2009.
- G. Anna et al., "ChEMBL: A large-scale bioactivity database for drug discovery," *Nucleic Acids Res.*, vol. 40, pp. D1100–D1107, 2012.
- M. Kanehisa, S. Goto, M. Hattori, K. F. Aoki-Kinoshita, and M. Hirakawa, "From genomics to chemical genomics: New developments in KEGG," *Nucleic Acids Res.*, vol. 34, pp. D354–357, 2006.
- S. Damian, S. Alberto, V. M. Christian, J. L. Juhl, B. Peer, and K. Michael, "STITCH 5: Augmenting protein-chemical interaction networks with tissue and affinity data," *Nucleic Acids Res.*, vol. 44, no. D1, pp. D380–D384, 2016.
- S. Ekins, "The next era: Deep learning in pharmaceutical research," *Pharmaceut. Res.*, vol. 33, no. 11, pp. 1–10, 2016.
- E. Gawehn, J. A. Hiss, and G. Schneider, "Deep learning in drug discovery," *Mol. Inform.*, vol. 35, no. 1, pp. 3–14, 2016.
- K. Tian, M. Shao, Y. Wang, J. Guan, and S. Zhou, "Boosting compound-protein interaction prediction by deep learning," *Methods*, vol. 110, no. 1, pp. 64–72, 2016.
- H. Öztürk, A. Özgür, and O. Ozkirimli, "DeepDTA: Deep drug-target binding affinity prediction," *Bioinformatics*, vol. 34, no. 17, pp. i821–i829, 2018.
- H. Öztürk, E. Ozkirimli, and A. Özgür, "WideDTA: Prediction of drug-target binding affinity," 2019, *arXiv:190204166*.
- M. Tsubaki, K. Tomii, and J. Sese, "Compound-protein interaction prediction with end-to-end learning of neural networks for graphs and sequences," *Bioinformatics*, vol. 35, no. 2, pp. 309–318, 2019.
- T. Nguyen, H. Le, T. P. Quinn, T. Nguyen, and S. Venkatesh, "GraphDTA: Predicting drug-target binding affinity with graph neural networks," *Bioinformatics*, vol. 37, no. 8, pp. 1140–1147, 2020.
- F. Scarselli, M. Gori, A. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, Jan. 2009.
- S. Kearnes, K. McCloskey, M. Berndl, V. Pande, and P. Riley, "Molecular graph convolutions: Moving beyond fingerprints," *J. Comput.-Aided Mol. Des.*, vol. 30, pp. 595–608, 2016.
- F. T. N. Kip and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:160902907*.
- C. Sun, P. Xuan, T. Zhang, and Y. Ye, "Graph convolutional autoencoder and generative adversarial network-based method for predicting drug-target interactions," *IEEE-ACM Trans. Comput. Biol. Bioinf.*, vol. 19, no. 1, pp. 455–464, Jan./Feb. 2020, doi: [10.1109/TCBB.2020.2999084](https://doi.org/10.1109/TCBB.2020.2999084).
- M. Karimi, D. Wu, Z. Wang, Y. Shen, and A. Valencia, "DeepAffinity: Interpretable deep learning of compound-protein affinity through unified recurrent and convolutional neural networks," *Bioinformatics*, vol. 35, no. 18, pp. 3329–3338, 2019.
- S. Zheng, Y. Li, S. Chen, J. Xu, and Y. Yang, "Predicting drug-protein interaction using quasi-visual question answering system," *Nature Mach. Intell.*, vol. 2, no. 2, pp. 134–140, 2020.
- L. Chen et al., "TransformerCPI: Improving compound-protein interaction prediction by sequence-based deep learning with self-attention mechanism and label reversal experiments," *Bioinformatics*, vol. 36, no. 16, pp. 4406–4414, 2020.
- Z. Cheng, C. Yan, F. Wu, and J. Wang, "Drug-target interaction prediction using multi-head self-attention and graph attention network," *IEEE-ACM Trans. Comput. Biol. Bioinf.*, early access, May 6, 2021, doi: [10.1109/TCBB.2021.3077905](https://doi.org/10.1109/TCBB.2021.3077905).

- [31] H.-T. Cheng *et al.*, "Wide & deep learning for recommender systems," in *Proc. 1st Workshop Deep Learn. Recommender Syst.*, 2016, pp. 7–10.
- [32] H. Guo, R. Tang, Y. Ye, Z. Li, and X. He, "DeepFM: A factorization-machine based neural network for CTR prediction," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, 2017, pp. 1725–1731.
- [33] S. Rendle, "Factorization machines," in *Proc. 10th IEEE Int. Conf. Data Mining*, 2010, pp. 995–1000.
- [34] H. Liu, J. Sun, J. Guan, Z. Jie, and S. Zhou, "Improving compound–protein interaction prediction by building up highly credible negative samples," *Bioinformatics*, vol. 31, no. 12, pp. i221–i229, 2015.
- [35] D. S. Wishart *et al.*, "DrugBank: A knowledgebase for drugs, drug actions and drug targets," *Nucleic Acids Res.*, vol. 36, no. suppl_1, pp. D901–D906, 2008.
- [36] S. Günther *et al.*, "SuperTarget and matador: Resources for exploring drug–target relationships," *Nucleic Acids Res.*, vol. 36, no. suppl_1, pp. D919–D922, 2008.
- [37] Y. Tabei and Y. Yamanishi, "Scalable prediction of compound–protein interactions using minwise hashing," *BMC Syst. Biol.*, vol. 7, no. Suppl 6, pp. S3–S3, 2013.
- [38] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [39] F. Costa and K. De Grave, "Fast neighborhood subgraph pairwise distance kernel," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 255–262.
- [40] C. G. Wermuth, C. R. Ganellin, P. Lindberg, and L. A. Mitscher, "Glossary of terms used in medicinal chemistry," *Pure Appl. Chem.*, vol. 70, no. 5, pp. 1129–1143, 1998.
- [41] Q. W. Dong and W. L. Lin, "Application of latent semantic analysis to protein remote homology detection," *Bioinformatics*, vol. 22, no. 3, 2006, Art. no. 285.
- [42] K. Kawashima, "AAindex: Amino acid index database," *Nucleic Acids Res.*, vol. 28, no. 1, 2000, Art. no. 374.
- [43] M. R. Zhang, J. Lucas, G. Hinton, and J. Ba, "Lookahead optimizer: K steps forward, 1 step back," *Proc. Adv. Neural Inform. Process. Syst.*, vol. 32, pp. 9597–9608, 2019.
- [44] L. Liu, H. Jiang, P. He, W. Chen, and J. Han, "On the variance of the adaptive learning rate and beyond," 2019, *arXiv:190803265*.
- [45] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," 2017, *arXiv:171010903*, doi: [10.48550/arXiv.1710.10903](https://doi.org/10.48550/arXiv.1710.10903).
- [46] W. Kühlbrandt, "Structure and mechanisms of F-Type ATP synthases," *Annu. Rev. Biochem.*, vol. 88, no. 1, pp. 515–549, 2019.
- [47] H. Shin. *et al.*, "Identification of ALDH6A1 as a potential molecular signature in hepatocellular carcinoma via quantitative profiling of the mitochondrial proteome," *J. Proteome Res.*, vol. 19, no. 4, pp. 1684–1695, 2020.



Chunyan Tang is currently working toward the PhD degree with the School of Computer Science and Technology, South China University of Technology, Guangzhou, Guangdong, China. She is also with an engineer with the School of Computer, Electronics and Information, Guangxi University, China. Her main research interests include bioinformatics, computer network, and machine learning.



Cheng Zhong received the PhD degree in computer science and technology from the University of Science and Technology of China, in 2003. He is with a professor with the School of Computer, Electronics and Information, Guangxi University, China, and an outstanding member of Chinese Computer Federation. He has published more than 100 journal and conference papers. His research interests include parallel computing, biological information computing, and social computing.



Mian Wang received the PhD degree in microbiology from Guangxi University, in 2017. She is now an Associate Professor with the College of Life Science and Technology, Guangxi University, China. Her research interests include the screening and mechanism study of antitumor drugs.



Fengfeng Zhou is a professor of health informatics with the College of Computer Science and Technology, and Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, P. R. China. His research interests include the development and optimization of feature selection and feature engineering algorithms for biomedical Big Data.

▷ For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/csdl.