ACS OMEGA

http://pubs.acs.org/journal/acsodf

Article

SS-GNN: A Simple-Structured Graph Neural Network for Affinity Prediction

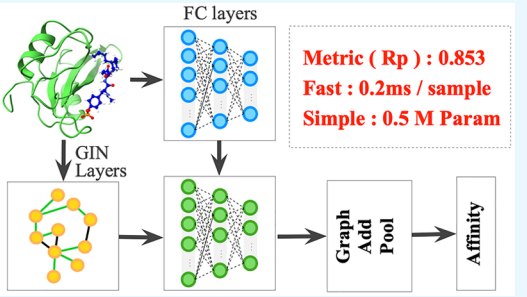
Shuke Zhang,[⊥] Yanzhao Jin,[⊥] Tianmeng Liu, Qi Wang, Zhaohui Zhang,* Shuliang Zhao,* and Bo Shan*

Cite This: *ACS Omega* 2023, 8, 22496−22507

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: Efficient and effective drug-target binding affinity (DTBA) prediction is a challenging task due to the limited computational resources in practical applications and is a crucial basis for drug screening. Inspired by the good representation ability of graph neural networks (GNNs), we propose a simple-structured GNN model named SS-GNN to accurately predict DTBA. By constructing a single undirected graph based on a distance threshold to represent protein−ligand interactions, the scale of the graph data is greatly reduced. Moreover, ignoring covalent bonds in the protein further reduces the computational cost of the model. The graph neural network-multilayer perceptron (GNN-MLP) module takes the latent feature extraction of atoms and edges in the graph as two mutually independent processes. We also develop an edge-based atom-pair feature aggregation method to represent complex interactions and a graph pooling-based method to predict the binding affinity of the complex. We achieve state-of-the-art prediction performance using a simple model (with only 0.6 M parameters) without introducing complicated geometric feature descriptions. SS-GNN achieves Pearson's R_p = 0.853 on the PDBbind v2016 core set, outperforming state-of-the-art GNN-based methods by 5.2%. Moreover, the simplified model structure and concise data processing procedure improve the prediction efficiency of the model. For a typical protein−ligand complex, affinity prediction takes only 0.2 ms. All codes are freely accessible at https://github.com/xianyuco/SS-GNN.

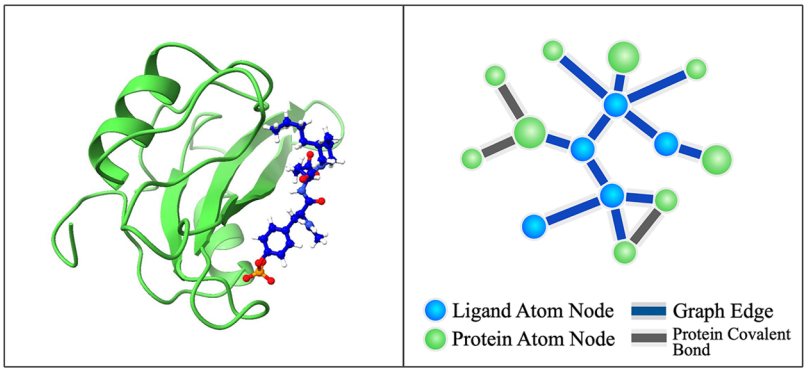


1. INTRODUCTION

Drug development is a process with long cycles, high investments, and high risks.[1,2] Drug-target binding affinity (DTBA) prediction plays an important role in drug development[3−6] and is also an important basis for drug screening. Accurate DTBA predictions will significantly reduce new drug development costs and speed up the drug discovery process,[7] which remain a challenge today. Traditional methods such as classical scoring functions (SFs)[8−11] do not estimate binding affinity well, and molecular dynamics (MD) simulations[12,13] have improved prediction accuracy, but they are too slow for large-scale applications. With the development of machine learning (ML), a large number of models for predicting drug-target interactions based on traditional ML methods[14−21] have emerged. ΔVinaRF$_{20}$ combines AutoDock Vina and random forest models to predict binding affinity. ECIF[22] introduces Extended Connectivity Interaction Features to describe protein−ligand complexes, combined with machine learning SFs to improve binding affinity prediction. AGL-Score[23] and DCML[24] are based on algebraic graph descriptors and the Dowker complex for molecular representation, respectively, and are trained via gradient boosted trees (GBT). HPC/HWPC,[25] PerSpect ML,[26] and TopBP[27] utilize topological descriptor-based ML methods to predict binding affinity. These ML models have achieved good results, they typically use well-designed manual features and require special domain knowledge and experience.

Deep learning (DL)-based methods can automatically extract features from available data. Therefore, DL-based methods have received increasing attention, and a large number of DL-based methods[7,28−31] have been proposed for binding affinity prediction, most of which have better performance and greater potential for capacity enhancement than traditional ML algorithms. Among them, the most commonly used methods are convolutional neural networks (CNNs) and graph neural networks (GNNs). With the increase in ligand-target 3D structural data, learning to predict binding affinity from 3D structural complexes has become a hot area of research. To encode the structural information on proteins and drugs as comprehensively as possible, some DL models based on 3D structure embedding[32−36] have been proposed. In OnionNet,[37] the contacts between proteins and ligands are grouped according to different distance ranges, and the resulting features are fed into a CNN. In OnionNet-2,[38] the contact logarithms between protein residues and ligand atoms are used as input features to the CNN model to predict

Received: February 2, 2023
Accepted: June 1, 2023
Published: June 15, 2023



(a) 3D Protein-Ligand Complex (b) Complex Interaction Graph

Figure 1. Graph representation of the protein−ligand complex. (a) 3D structure of the complex. (b) Graph representation ignoring protein atoms outside the threshold and all covalent bonds in the protein.

binding affinity. In K_{Deep},[39] FAST,[40] and Pafnucy,[35] 3D grids are applied to represent protein−ligand complexes, and 3D CNNs are applied to generate feature embeddings. Furthermore, Nguyen et al. propose to describe biomolecular structures through mathematical representations.[41,42] TNET-BP[43] combines topological and convolutional neural networks to achieve state-of-the-art predictive performance in affinity prediction. Mol-PSI[44] uses 2D images to represent the molecular structures and interactions and combines the CNN model to improve the affinity prediction accuracy.

In order to better represent the structural features of molecular graph, GNNs with good representation ability are used for DTBA prediction.[32,45−51] In GNN-based models, graph structures are applied to represent atoms and their covalent bonds, and GNNs are applied to predict drug-target binding affinity. Some new methods have been proposed that consider the spatial information on the relative positions of atoms between ligands and proteins to improve GNN-based DTBA prediction models. The SGCN model[52] proposed by Danel et al. considers the spatial information on the nodes. SIGN[53] proposed by Li et al. introduces polar coordinates and considers angle information and the case of long-range interactions between atoms. The IGN model proposed by Jiang et al.[54] encodes the chemical and structural information in 3D space into a molecular graph, which comprehensively represents protein−ligand interaction patterns and adopts three molecular graphs to represent complexes. The model has good performance on the PDBbind data set. The MP-GNN[55] proposed by Li et al. is a multiphysics graph neural network model that exhibits good predictive ability in predicting SARS-CoV/SARS-CoV-2 inhibitor complexes.

The graphDelta[56] model applies a multitask learning method based on graph convolutional neural networks. The AweGNN[57] model is a neural network model based on geometric features, which can be automatically parametrized, and the model can achieve advanced performance of molecular property prediction. PIGNet[58] uses physics-informed equations to instruct model learning. By utilizing the diversity of protein−ligand chemistry and structure for data augmentation, the generalization ability of the model is further improved. PIGNet performs well in both prediction accuracy and stability. The MGraphDTA[59] model utilizes a multiscale graph neural network with 27 graph convolutional layers to predict the affinity, which can simultaneously capture the local and global structures of the compound, and also uses the gradient weighted affinity activation mapping method for visual explanation. GIGN[60] model uses a graph neural network to predict protein−ligand binding affinity, considering the 3D structures of the complexes and protein−ligand physical interactions. GIGN achieves state-of-the-art performance. GNN-based frameworks have made good progress in binding affinity prediction, but most of these frameworks employ complicated model architecture and intricate geometric structures data that complicate the model. They are still not well suited for large-scale application in engineering. Therefore, it is highly desirable to develop an efficient DTBA prediction model with simple model architecture to meet the requirement of high efficiency.

To tackle the above problems, in this paper, we develop a novel method to improve the DTBA prediction model based on a GNN named SS-GNN. Compared with the state-of-the-art methods, it not only achieves good prediction performance but also has a simple structure and high prediction efficiency. SS-GNN is equipped with three modules to accomplish affinity prediction. We apply a single undirected graph to represent protein−ligand complexes, where nodes are atoms and edges are the interactions of atoms (Figure 1). The appropriate distance threshold is obtained by the hyperparameter tuning. We use a smaller threshold while ensuring model performance. A smaller threshold will reduce the number of nodes and edges in a complex graph, and reduce the scale of the graph fed to the model. We design a hybrid feature extraction module (GNN-MLP) to extract useful features for atoms and interactions, respectively, and implement a lightweight feature embedding process via a two-layer graph isomorphism network (GIN) submodule and a three-layer multilayer perceptron (MLP) submodule. By aggregating the embedding information on each edge and its connected atom pairs, edge-based atom-pair aggregation features can be obtained, and by applying a simple MLP, the binding affinity of a single edge can be predicted. Finally, by summing the individual edge affinity predictions by employing a graph pooling module, the affinity of the complex can be obtained. In summary, the main contributions of our work are as follows:

(1) Protein−ligand complex representation based on a single undirected graph. The distance threshold is selected by an experimental approach based on cross-validation. And the covalent bonds in the ligands are preserved while the covalent bonds in the proteins are ignored. The model achieves the best trade-off between prediction accuracy

https://doi.org/10.1021/acsomega.3c00085
ACS Omega 2023, 8, 22496−22507

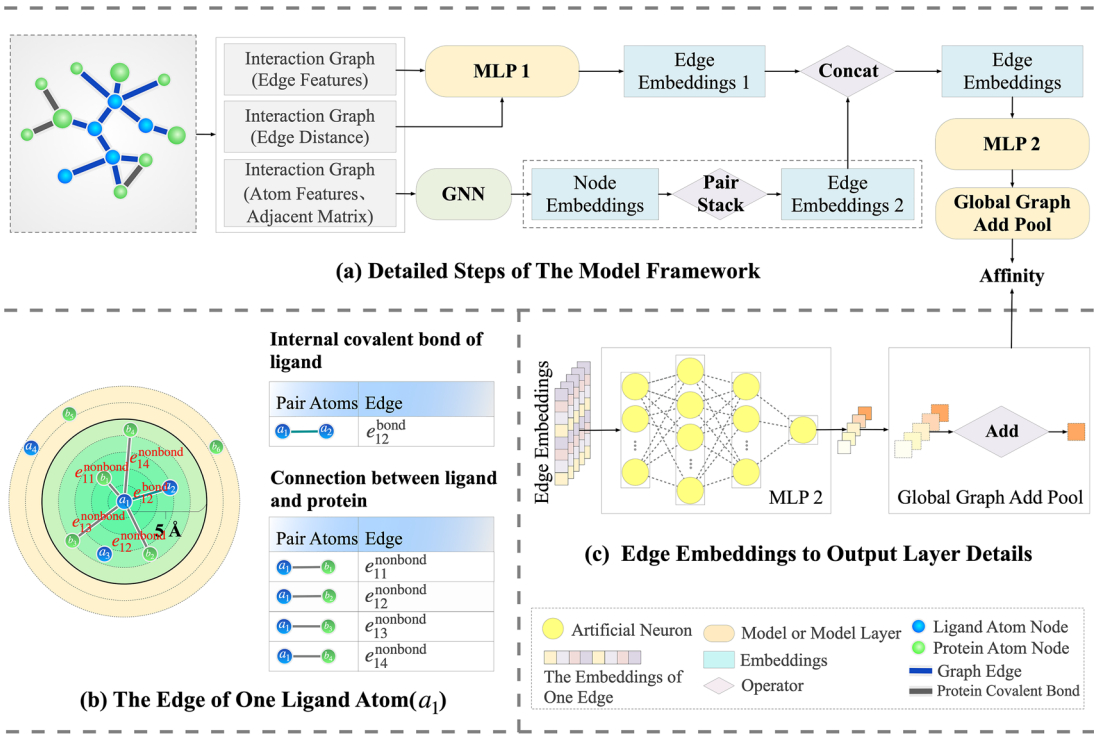


Figure 2. (a) Detailed steps of the SS-GNN framework. The SS-GNN takes a graph representation of drug−protein complexes as input and the prediction of binding affinity as output. (b) The two types of edges connected to an example ligand atom. (c) Details of the affinity prediction module.

and computational complexity. The discretization of the interatomic distance improves the computational efficiency and generalization ability to a certain extent and further improves the performance of the model.

(2) Hybrid feature extraction based on GNN-MLP. We regard the feature extraction of atoms and edges in the graph as two independent processes: the atom features are extracted by applying a simple and effective two-layer GIN, and the edge features are extracted by applying a lightweight MLP. Moreover, the single undirected graph representation not only simplifies the model but also makes updating the node information on proteins and ligands in the GNN more efficient.

(3) Edge-based atom-pair feature aggregation and graph pooling-based affinity prediction. The embedding vectors of each edge and its connected atom pairs are concatenated to achieve feature aggregation and form the inputs of the affinity prediction module. The predicted outputs of all individual edges are summed through a graph pooling layer to obtain the binding affinity of the complex.

Unlike other models, our data processing procedure avoids the high complexity caused by extracting complicated geometric structures. As a result, the number of parameters in the entire model is only 0.6 M. The simplicity of the model and data processing procedure leads to a simple and low-complexity SS-GNN. Experiments demonstrate the effectiveness and efficiency of the proposed model. In section 2, we introduce the detailed model architecture of SS-GNN. In section 3, we present the experimental results and compare them with those of state-of-the-art methods in similar tasks. Finally, section 4 summarizes our proposed method and briefly describes our future research plans.

2. SS-GNN

In this section, we introduce the proposed SS-GNN method. The SS-GNN defines the prediction of DTBA as a regression task, in which the model's input is the drug-target representation, and the output is a continuous value representing the binding affinity score between the drug and the target protein. The overall architecture of the SS-GNN is shown in Figure 2. Our approach consists of graph representation of complexes based on the distance threshold, hybrid mode feature extraction, feature aggregation, and affinity prediction. We first give an overview of the SS-GNN considering the 3D structure of the protein−ligand complex. In the following subsections, we elaborate the key modules.

2.1. Protein−Ligand Complex Representation Based on a Single Undirected Graph. Given a protein−ligand complex as shown in Figure 1(a), it can be described by the graph of interactions between atoms. As a rule of thumb, when the distance between protein−ligand atom pairs is greater than a certain threshold, the interactions between them do not contribute much to the interactions of the overall complex. In the initial stage of the experiment, we have constructed the complex graph using ligand atoms as well as all protein atoms and achieved good results. However, the number of atoms in a protein is much larger than that in a ligand. To verify whether the ligand features are overwhelmed by excessive protein features, we remove all the features of the ligand and replace them with random numbers. Experiments show that the model is still valid to a certain extent, which led us to think about how to make better use of ligand features and whether all protein atoms are required in the model. To this end, we propose a distance-threshold-based graph representation method that employs the ligand and its partial protein to construct a complex interaction graph. We set the distance threshold as an

https://doi.org/10.1021/acsomega.3c00085
ACS Omega 2023, 8, 22496−22507

optimizable hyperparameter and experimentally determine a feasible value.

Unlike some other GNN-based DTBA prediction methods, our proposed SS-GNN applies only one single protein−ligand complex graph $\mathcal{G}$ to characterize the interactions of the complex instead of building ligand and protein graphs separately. We first define the atom node set of the ligand as $\mathcal{V}^L$. We also define the atom node set of the protein as

$$\mathcal{V}^P = \{a_j | a_j \text{ is a protein atom satisfying } d(a_i, a_j) \le \theta, \text{ where } a_i \in \mathcal{V}^L\}$$

where $d(a_i, a_j) = \left\|\mathbf{p}_i - \mathbf{p}_j\right\|_2$, $\mathbf{p}_i$ and $\mathbf{p}_j$ denote the coordinates of a_i and a_j, and θ is a hyperparameter representing distance threshold. Then, we define the protein−ligand complex as an undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \mathcal{V}^L \cup \mathcal{V}^P$ is the node set and $\mathcal{E}$ is the edge set containing two types of edges formed by atoms in $\mathcal{V}$, protein−ligand interactions and covalent bonds between ligand atoms.

We introduce a distance threshold θ that can significantly reduce the size of the graph. Furthermore, we do not employ covalent bonds within the protein, which also greatly reduces the number of edges in the complex graph. Then, we number the ligand atoms and the retained protein atoms, and construct the corresponding adjacency matrix $\mathbf{A} = [\mathbf{A}_{ij}]_{\mathcal{N}^{\mathcal{V}} \times \mathcal{N}^{\mathcal{V}}}$ where $\mathbf{A}_{ij}$ is defined as eq 1.

$$\mathbf{A}_{ij} = \begin{cases} 1, & a_i, a_j \in \mathcal{V}^L \text{ and there is a covalent bond between } a_i \text{ and } a_j \text{ and } i \neq j \\ 1, & a_i \in \mathcal{V}^L, a_j \in \mathcal{V}^P \text{ and } d(a_i, a_j) \le \theta \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

By introducing the adjacency matrix representing both bond interactions and atomic nonbonded interactions, our model can learn how protein−ligand interactions affect the node features of each atom.

In the graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, each node includes 11 features, each feature is represented by a vector, and these vectors are concatenated to form the initial feature vector of a node. Let $\mathbf{x}_i$ denote the initial feature vector of node a_i in $\mathcal{V}$, the initial feature vectors of all nodes in $\mathcal{V}$ form the vector set as $\mathbf{x} = \{\mathbf{x}_i | a_i \in \mathcal{V}\}$. The types of edges include the covalent bonds between ligand atoms and protein−ligand interactions. The features of covalent bonds include the covalent bond type, whether the bond is in a ring, bond length, bond direction, and bond stereochemistry. To ensure that the dimensions of the two types of edges are consistent, the features of protein−ligand interactions are the same as those of covalent bonds, the bond length is the distance between two atoms, and other features take default values. In addition, two different types of edges are embedded in features using 0−1 codes to distinguish them. All features of an edge are encoded as vectors and concatenated to form the initial feature vector of an edge. Let $\mathbf{e}_{ij}$ denote the initial feature vector of edge e_{ij} in $\mathcal{E}$, the initial feature vectors of all edges in $\mathcal{E}$ form the vector set as $\mathbf{e} = \{\mathbf{e}_{ij} | e_{ij} \in \mathcal{E}\}$. A list of initial features for nodes and edges is summarized in Table 1.

It is worth noting that the bond length of edge e_{ij} in $\mathcal{E}$ is the Euclidean distance calculated based on the 3D coordinates of both atom nodes a_i and a_j, which is a continuous real value denoted by $d(a_i, a_j)$. To further simplify the computation and improve the model performance, we discretize the distance as shown in eq 2.

$$\hat{d}(a_i, a_j) = \lfloor d(a_i, a_j) \rfloor \quad (2)$$

where $\hat{d}(a_i, a_j)$ denotes the value after discretization. SS-GNN applies the single undirected graph representation method based on the distance threshold, which greatly reduces the amount of computation and makes the model more lightweight.

Table 1. List of the Initial Features of Nodes and Edges

name	description
	Node Features
atom type	B, C, N, O, S, P, Se, halogens, metals, other
atom charge number	formal charge for an atom. range:[−5,5], other
hybridization	S, SP, SP2, SP3, SP3D, SP3D2, other
atom valence	range:[0,7], other
atom degree	total number of bonded atom neighbors range:[0,10], other
number of hydrogens	explicit and implicit hydrogens. range:[0,8], other
atom coordinates	position coordinates of atoms in 3D space
chirality	unspecified, tetrahedral_CW, tetrahedral_CCW, other
atomic mass	mass of a single atom
aromatic	whether if the atom is aromatic. 0 or 1
belongs to the protein	whether the atom belongs to the protein, 0 or 1
	Edge Features
covalent bond type	single, double, triple, aromatic, unspecified, zero, other
aromatic	whether the bond is in an aromatic ring. 0 or 1
bond length	distance between connected atoms in 3D space
bond direction	none, endupright, enddownright, eitherdouble, unknown
bond stereochemistry	stereonone, stereoany, stereoz, stereoe, stereocis, stereotrans
edge type	protein−ligand interaction or a bond between ligand atoms. 0 or 1

2.2. Hybrid Feature Extraction Based on GNN-MLP.

Different from other methods, we propose a hybrid feature extraction module named GNN-MLP to extract the features of the complexes. These two modules are independent of each other; the GNN-based network is applied to learn the latent features of atoms, and a multilayer perceptron (MLP) is applied to learn the latent features of edges. Each feature extraction module is very simple and lightweight.

2.2.1. Node Feature Extraction Based on GIN. Xu et al. developed a simple and powerful graph learning method, the graph isomorphism network (GIN), and theoretically proved that the model has the maximum discriminant ability in GNNs.[61] We utilize a GIN-based module to learn the node representations of the protein−ligand complex.[45,62−67] Given the adjacency matrix $\mathbf{A}$ of graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ and the initial feature vectors $\mathbf{x}$ of all nodes in $\mathcal{V}$. First, the adjacency matrix $\mathbf{A}$ and the initial features $\mathbf{x}$ are fed into the GIN module; then, the representation vector for each node is updated by aggregating information from its neighboring nodes based on adjacency matrix $\mathbf{A}$; and finally, iterative message passing is employed to extract the latent representations of all nodes. Since the composition of a function can be represented by

https://doi.org/10.1021/acsomega.3c00085
ACS Omega 2023, 8, 22496−22507

multilayer perceptions (MLPs), the MLP method is applied to update all node features in each GIN layer. For any node a_i in $\mathcal{V}$, the k-th GIN layer updates its feature representation as eq 3.

$$\mathbf{x}_i^{(k)} = \mathrm{MLP}^{(k)}\left((1 + \varepsilon^{(k)})\mathbf{x}_i^{(k-1)} + \sum_{j \in \mathcal{N}(i)} \mathbf{x}_j^{(k-1)}\right) \tag{3}$$

where ε is either a learnable parameter or a fixed scalar, $\mathcal{N}(i)$ is a set of nodes adjacent to node a_i, and $\mathbf{x}_i^{(k)}$ is the updated feature vector of node a_i. $\mathbf{x}_i^{(0)}$ is initialized as $\mathbf{x}_i$. By feeding the adjacency matrix $\mathbf{A}$ and the initial feature vectors $\mathbf{x}$ into the GIN module, we can get the extracted latent feature vectors of the complex atoms at the module's output as $\mathbf{x}' = \mathrm{GIN}(\mathbf{x},\mathbf{A})$.

In our proposed method, only a single graph of protein−ligand complexes is fed into the GIN network, resulting in less input data, thereby reducing model computation. Furthermore, the GIN module consists of two GIN layers, each of which is followed by a batch normalization layer to speed up the training. Only two GIN layers are applied in this module, resulting in a relatively lightweight model with only 0.039 M parameters.

2.2.2. Edge Feature Extraction Based on MLP. In this part, we utilize the MLP1 module (Figure 2(a)) for edge feature extraction. This MLP-based module consists of three fully connected layers, where each of the first two layers is followed by a ReLU activation function for nonlinear transformation. The MLP1 module is designed to learn the edge features of the protein−ligand complex, which include two types of edges: covalent bonds inside the ligand and edges connecting protein and ligand atoms. Given the initial feature vector $\mathbf{e}_{ij}$ for any edge e_{ij} in $\mathcal{E}$, the extracted edge embedding vector $\mathbf{e}'_{ij}$ can be obtained from the module's output: $\mathbf{e}'_{ij} = \mathrm{MLP1}(\mathbf{e}_{ij})$.

2.3. Feature Aggregation and Affinity Prediction. *2.3.1. Edge-Based Atom-Pair Feature Aggregation.* To well represent the interactions in the complex, we propose an edge-based atom-pair feature aggregation module. To get the aggregated features for any edge e_{ij} in $\mathcal{E}$, we first concatenate both feature vectors $\mathbf{x}'_i$ and $\mathbf{x}'_j$ of its connected atom pair a_i and a_j as eq 4, and then in eq 5 we integrate the concatenated feature vector $\mathbf{x}_{ij}$ with the edge embedding vector $\mathbf{e}'_{ij}$ obtained by the MLP1 module.

$$\mathbf{x}_{ij} = \mathbf{x}'_i \| \mathbf{x}'_j \tag{4}$$

$$\mathbf{AGG}_{ij} = \mathbf{e}'_{ij} \| \mathbf{x}_{ij}, \ \forall\ e_{ij} \in \mathcal{E} \tag{5}$$

where $\mathbf{x}'_i$ and $\mathbf{x}'_j$ denote latent feature vectors of the atom pair a_i and a_j obtained through the GIN module, and $\|$ is a concatenation of two vectors. Finally, the feature vector $\mathbf{AGG}_{ij}$ can be interpreted as the final information on aggregated features and is directly delivered to the followed affinity prediction module.

2.3.2. Graph Pooling-Based Affinity Prediction. As shown in Figure 2(c), we utilize the MLP2 module and the graph pooling module[45,68−71] for an affinity prediction. The MLP2 module is a 4-layer feedforward neural network; except for the output layer, each layer is followed by a ReLU activation function for nonlinear transformation. To predict protein−ligand complex binding affinity with graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, we first predict the output value of each edge in $\mathcal{E}$. For any edge e_{ij} in $\mathcal{E}$, by feeding the aggregated feature vector $\mathbf{AGG}_{ij}$ into MLP2, the corresponding output value $y(e_{ij})$ can be obtained in eq 6. Then the output values of all edges in $\mathcal{E}$ form the output vector $\mathbf{y}$ for graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$. Finally, by passing the vector $\mathbf{y}$ to the graph pooling module, the predicted binding affinity $\hat{y}$ of the complex can be estimated as eq 7.

$$y(e_{ij}) = \mathrm{MLP2}(\mathbf{AGG}_{ij}), \quad \forall\ e_{ij} \in \mathcal{E} \tag{6}$$

$$\hat{y} = \mathrm{ADDPOOL}(\mathbf{y}) = \sum_{e_{ij} \in \mathcal{E}} y(e_{ij}) \tag{7}$$

As mentioned above, each module in SS-GNN adopts a concise network structure. Table 2 shows the size of each module in the model of SS-GNN. In general, a simplified single graph representation method based on distance threshold selection and a simple feature extraction process lead to a lightweight model.

Table 2. Size of Each Module in the SS-GNN

Network module	GIN	MLP1	MLP2
network layers	2	3	4
parameters	0.039 M	0.003 M	0.526 M

2.4. Loss Function. In this end-to-end model SS-GNN, we treat the affinity prediction as a regression task. Given a train set D with N samples, the predicted value and the truth value for any sample i are $\hat{y}_i$ and y_i, respectively. The training objective is to minimize the mean-squared-error (MSE) loss defined as eq 8.

$$\text{MSE loss} = \frac{1}{N}\sum_{i=1}^{N} (\hat{y}_i - y_i)^2 \tag{8}$$

3. RESULTS AND DISCUSSION

In this section, we conduct a comprehensive experimental evaluation of the recent PDBbind v2016 and v2013 core sets to explain the benefits of exploiting the proposed model in affinity prediction. In the following subsections, we first introduce the distance threshold selection, analyze our proposed model through extensive ablation studies and then report experimental comparisons with recently proposed state-of-the-art methods. Finally, we present a detailed discussion of our experiments and provide useful insights and conclusions.

3.1. Data Sets and Evaluation Protocols. To evaluate the performance of our proposed method, we adopt the widely used benchmark PDBbind data set v2019.[72,73] This data set is a well-known public data set used to predict DTBA, and is a comprehensive database composed of 3D structure data of drug targets. This data set provides 3D structures of protein−ligand complexes and the corresponding binding affinity represented by pK_a values determined experimentally. It includes three overlapping subsets, namely, the general set U_g, the refined set U_r and the core set U_c, where $U_c \subset U_r \subset U_g$. The general set contains all samples of the data set, while the refined set is a subset with higher quality data selected from the general set. The core set is designed as the highest quality benchmark and is often used as a test set. The protein−ligand complexes in the core set have high-quality crystal structures and reliable experimental affinity data.

In this paper, we employ two test sets (the v2016 and v2013 core sets) to test the performance of SS-GNN. The v2016 core set[74] contains 285 structurally diverse ligand−receptor

https://doi.org/10.1021/acsomega.3c00085
ACS Omega 2023, 8, 22496−22507

complexes (270 samples are used for testing, and 15 samples fail in the reading and processing of protein or compound structure information). The v2013 core set[75] contains 195 complex samples (189 samples are used for testing, and 6 samples fail in the reading and processing of protein or compound structure information).

For the experiments with the v2016 core set, we remove the overlapping part of the corresponding core set from the refined set, and the remaining samples are employed for model learning, of which 90% are used as the training set (4,073 samples) and 10% are used as the validation set. Moreover, we carry out a supplementary experiment on the larger general set to further analyze the generalizability of our model. Samples in the general set that overlap with the core set are removed. Similar to the process with the refined set, 90% of the remaining samples are used as the training set (15,394 samples), and 10% are used as the validation set. The processing of the experimental data set for the v2013 core set is the same as that of the v2016 core set. In the end, 4,005 training samples are obtained in the refined set, and 15,317 training samples are obtained in the general set.

For model evaluation, we follow previous work to evaluate performance from different perspectives using two main indicators: root mean squared error (RMSE)[76] and Pearson correlation coefficient (R_p). In addition, to achieve a more diverse evaluation, the concordance index(CI),[77] coefficient of determination (R^2), and mean absolute error (MAE) are calculated. The smaller the values of RMSE and MAE and the larger the values of R_p, R^2, and CI are, the better is the performance of the model.

3.2. Implementation Details. We implement our approach based on the PyTorch toolbox. Experimentally, we apply the Adam optimizer and set the learning rate to 0.001. We train our network for 1000 epochs, and in each epoch, the training set is randomly divided into 192 mini-batches. Modeling experiments and benchmarking are carried out on a machine with an Intel(R) Xeon(R) CPU E5-2678 v3 @ 2.50 GHz CPU and an NVIDIA GeForce RTX 2080Ti graphics card.

3.3. Distance Threshold Selection Based on Cross-Validation. In this subsection, we introduce the method of distance threshold selection and verify the necessity of distance threshold selection. We choose the refined set as the training set and perform 5-fold cross-validation separately for different thresholds ranging from 4 to 8 Å. Table 3 shows the R_p values

Table 3. Cross-Validation Experimental Results with Different Thresholds

threshold	4 Å	5 Å	6 Å	7 Å	8 Å
$R_p(\uparrow)$	0.674 ± 0.033	0.717 ± 0.021	0.716 ± 0.026	0.711 ± 0.022	0.710 ± 0.026

of the model under different thresholds on the PDBbind v2019 refined set. The experimental results show that the model performs the worst when the threshold is 4 Å, while the difference in R_p is not significant when the threshold is 5 Å and larger.

To verify the effect of threshold selection on the size of the constructed complex graph, we calculate the average values of the number of atoms and edges for the 285 complex samples in the v2016 core set at different thresholds, as shown in Figure 3. As in common practice, all water molecules and hydrogen atoms in the PDB structures are removed. The number of ligand atoms and the intramolecular edges do not change with increasing distance threshold. With an increase in threshold, the number of protein atoms increases linearly, and the number of edges between ligands and proteins also rises dramatically. Figure 4 shows the number of protein covalent bonds for 50 samples randomly selected from the 285 samples in the PDBbind v2016 core set.

The number of protein covalent bonds increases significantly with increasing distance threshold.

Initially, we tried to use all covalent bonds between ligand atoms and covalent bonds between protein atoms within 5 Å, and the R_p on the PDBbind v2016 core set is 0.854 (Table S7). Since we would pay more attention to the interactions between ligand atoms and protein atoms, we consider removing the covalent bonds between protein atoms, the R_p on the core set of PDBbind v2016 is 0.853. It was shown that covalent bonds between protein atoms contribute little to the prediction of binding affinity. However, the number of covalent bonds between protein atoms is very large (Figure 4); therefore, our model does not use any covalent bonds between protein atoms, which is obviously different from other models. Ignoring the covalent bonds between protein atoms will greatly reduce the number of edges of the graph in the model input, making the scale of the graph smaller and improving the model efficiency. Figure 5 depicts the number of ligand−protein connections for 50 randomly selected samples. The average number of connections at a distance threshold of 5 Å is reduced to 1/6 of that at 8 Å. Considering the prediction performance and computational cost, we finally select 5 Å as the distance threshold. The subsequent experiments are carried out under the 5 Å threshold. Compared with other methods, SS-GNN applies a complex graph representation method based on a distance threshold, resulting in a substantial reduction in the size of the graph, which fully illustrates the computational advantages of SS-GNN.

To verify the validity of the ligand covalent bonds kept in the model, we removed all covalent bonds, leaving only noncovalent bonds between protein atoms and ligand atoms. We use the general set as the training set, and test the performance of the new model on the core set v2016. The results show that after removing all covalent bonds from the model, the performance (average RMSE = 1.460, average R_p = 0.784) decreased compared to the model with ligand covalent bonds retained (average RMSE = 1.181, average R_p = 0.853). This illustrates that incorporation of ligand covalent bonds could make the prediction performance of the model better. The experiment is also repeated 5 times with different random seeds. The detailed results are shown in Table S6.

To further illustrate the efficiency of SS-GNN, we test the model forward propagation runtime on the PDBbind v2019 refined set. When the threshold is 8 Å, the average prediction time per sample is 0.7 ms, and when the threshold is 5 Å, the average prediction time per sample is 0.2 ms. The lightweight model architecture and concise data processing procedure result in an efficient model.

3.4. Ablation Studies. To better understand the contribution of each component in the model to the overall performance, we remove each component from the model and conduct the ablation experiments using the PDBbind v2016 refined set, and the results are shown in Table 4.

First, we evaluate the usage of MLP1 in the GNN-MLP module, which is a fully connected neural network for learning

https://doi.org/10.1021/acsomega.3c00085
ACS Omega 2023, 8, 22496−22507

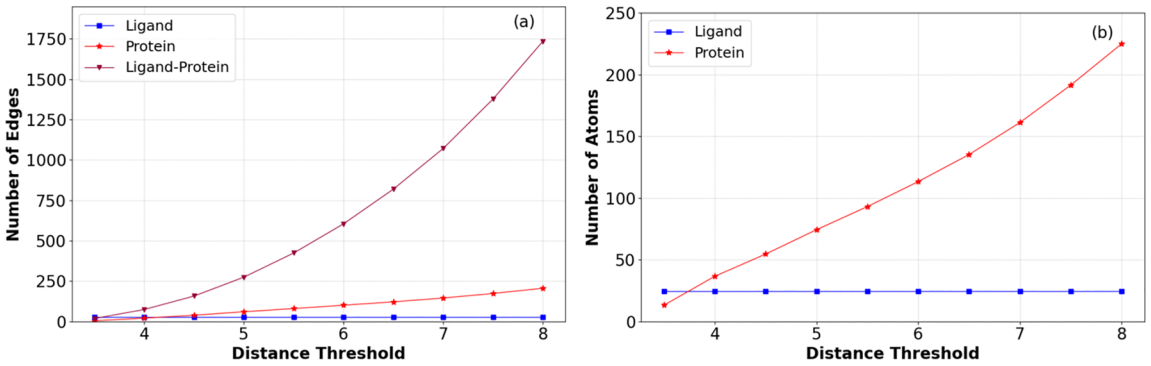


Figure 3. Average values of the number of edges (a) and atoms (b) for the 285 samples in the v2016 core set at different thresholds.

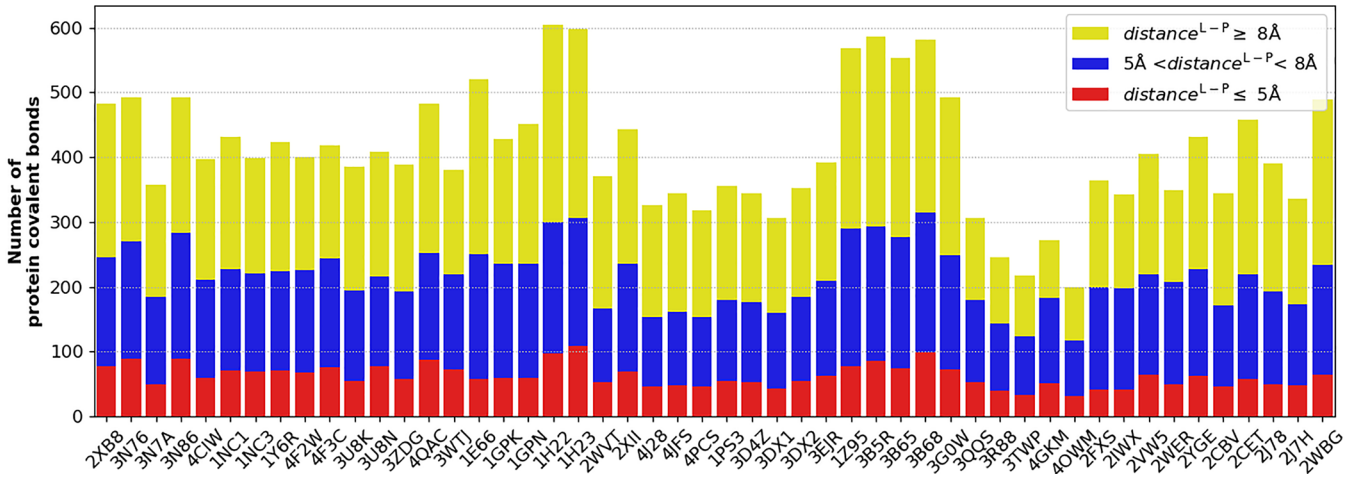


Figure 4. Number of covalent bonds formed by atoms that satisfy the threshold condition in the protein, where distance$^{L\text{-}P}$ is the distance between ligand atoms and protein atoms.

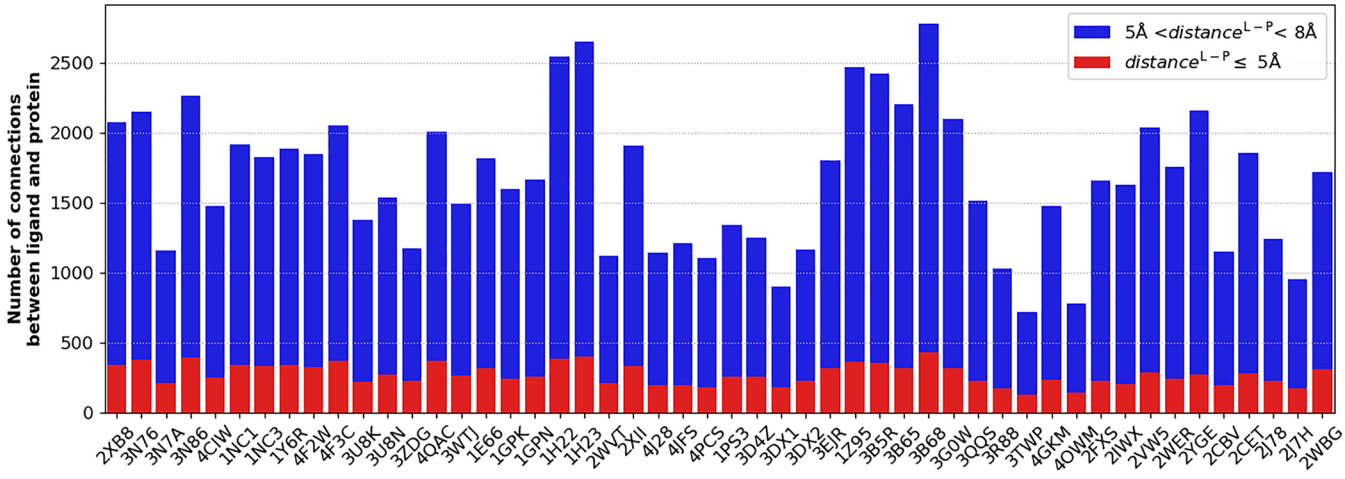


Figure 5. Number of connections between ligand atoms and protein atoms satisfying a threshold condition, where distance$^{L\text{-}P}$ is the distance between ligand atoms and protein atoms.

the latent features of edges. Compared with SS-GNN$_{\text{remove MLP1}}$ (with MLP1 module removed), SS-GNN has a 0.9% increase in R_p, a 0.3% decrease in RMSE, and a 0.7% increase in CI. This shows that the MLP1 module can learn the latent representation of the interactions between atom pairs at a deeper level and improve the model performance to a certain extent.

Second, we evaluate the effect of discretization of the distances between atoms (DDA) on the model performance. Compared with the model SS-GNN$_{\text{remove DDA}}$ without distance discretization, the RMSE of SS-GNN is reduced by 1.1%, R_p is improved by 0.5%, and CI is improved by 0.6%. The results show that the DDA module is necessary for SS-GNN.

Finally, we evaluate the effect of the number of layers of GIN. SS-GNN using a two-layer GIN shows an advantage over the model SS-GNN$_{\text{one-layer GIN}}$ using one-layer GIN (RMSE is reduced by 0.3%, R_p is improved by 0.5% and CI is improved by 0.5%); however, the improvement is minor. Moreover, SS-

https://doi.org/10.1021/acsomega.3c00085
ACS Omega 2023, 8, 22496−22507

Table 4. Experimental Results Showing the Effect of Different Components on the Model. In Each Table Cell, the Mean Value over Five Runs Is Reported as Well as the Standard Deviation

architecture	RMSE (↓)	R_p (↑)	CI (↑)	R^2 (↑)	MAE (↓)
SS-GNN	**1.181 ± 0.047**	**0.853 ± 0.012**	**0.833 ± 0.006**	**0.701 ± 0.024**	**0.920 ± 0.035**
SS-GNN$_{\text{remove MLP1}}$	1.184 ± 0.035	0.845 ± 0.011	0.827 ± 0.005	0.700 ± 0.018	0.927 ± 0.028
SS-GNN$_{\text{remove DDA}}$	1.194 ± 0.052	0.849 ± 0.006	0.828 ± 0.005	0.694 ± 0.027	0.926 ± 0.048
SS-GNN$_{\text{one-layer GIN}}$	1.185 ± 0.046	0.849 ± 0.011	0.829 ± 0.005	0.699 ± 0.024	0.928 ± 0.045
SS-GNN$_{\text{three-layer GIN}}$	1.220 ± 0.018	0.838 ± 0.007	0.825 ± 0.005	0.682 ± 0.010	0.942 ± 0.026

GNN outperforms SS-GNN$_{\text{three-layer GIN}}$ using a three-layer GIN (RMSE is reduced by 3.2%, R_p is improved by 1.8%, and CI is improved by 1.0%), indicating that increasing the number of GIN layers does not always lead to better performance of the model.

3.5. Comparison with the State of the Art. In this subsection, we first test our model on the PDBbind v2016 core set and then compare the proposed approach with other state-of-the-art methods on two data sets.

Experiments on the PDBbind Core Set. We employ the general set and refined set in PDBbind for model training and test the model on the PDBbind v2016 core set. The results are shown in Table 5.

All experiments in this paper are repeated 5 times with different random seeds. Each random seed represents a random shuffle of the data set. In each experiment, 90% of the samples are randomly selected as the training set, and the remaining 10% are selected as the validation set for model selection. We finally take the mean and standard deviation of the results of five independent experiments as the result of the average model SS-GNN$_{\text{average}}$ and take the model result with the largest R_p value as the result of the best model SS-GNN$_{\text{best}}$. For the PDBbind v2016 core set, the R_p of the best model trained on the refined set reaches 0.832, and that of the average model is 0.822; for the general set, the R_p of the best model reaches 0.870, and that of the average model is 0.853. The model achieves good performance on the refined set with a small sample size. Nonetheless, with the expansion of the training data set, the performance of the model is greatly improved, which further expands the prediction advantage. For the PDBbind v2013 core set, the R_p of the average model trained on the general set reaches 0.816.

To better represent the findings, the predicted binding affinities obtained using the PDBbind v2016 core set are shown in Figure 6, which presents the test results for the best models trained on the general and refined sets based on the PDBbind v2016 core set. The predicted values are highly correlated with the ground truth values. To ensure the stability of model prediction performance, 5 different random seeds are used in the model experiments in this paper.

Comparison with the State-of-the-Art Methods. We compare our proposed method with state-of-the-art methods.[19,22−27,35,37−40,44,53−56,74] Table 6 compares the results of our proposed SS-GNN with those of the state-of-the-art methods for the PDBbind core set v2016. SS-GNN ranks second only to TopBP on the general set, which shows that our lightweight structure can effectively learn deep features of the interactions of protein−ligand complexes. We select the chemical and biological attributes that can well represent the atom information during the data processing procedure and introduce an edge-based atom-pair feature aggregation module, which can better represent the interactions between atoms. We further utilize a GIN-based network and an MLP to learn the latent features of nodes and edges, respectively, in the complex graph. Therefore, despite the low number of atoms and interactions employed by SS-GNN, the model still achieves good performance. As the amount of training data increases, our model can provide more accurate predictions. Table 7 further demonstrates the comparison between SS-GNN and state-of-the-art GNN-based models. Our proposed model achieves good performance with concise graph representation and simple model architecture.

Models based on persistent homology and topological descriptors[23−27,44] achieve better results on smaller data sets. They rely more on expert knowledge and can achieve excellent results with reasonable feature extraction. There are also some DL-based models that achieve good results on the PDBbind v2016 core set, the best model of OnionNet-2[38] achieved R_p of 0.864 (16,626 training samples, 285 test samples), the best model of graphDelta[56] achieved an R_p of 0.870 (8,766 training samples, 285 test samples). In addition, the best model of ECIF::GBT[22] based on traditional descriptors and ML achieved R_p of 0.866 (9,299 training samples, 285 test samples), and the best model of our SS-GNN achieved R_p of 0.870. We also test the efficiency of HPC/HWPC on the PDBbind v2019 refined set. The average prediction time per sample is 1.2×10^4 ms (implemented with our optimized code which is orders of magnitude faster than the original implementation), while SS-GNN only needs 0.2 ms. The feature extraction process of HPC/HWPC is complicated, computationally intensive and slow. Due to the lack of large-scale standard data sets, our proposed model has not been tested on very large-scale data sets, but its superiority in accuracy and efficiency makes it more suitable for large-scale molecular docking tasks.

Table 5. Results of PDBbind Dataset Experiments

type	test set	training set	RMSE (↓)	R_p (↑)	CI (↑)	R^2 (↑)	MAE (↓)
SS-GNN$_{\text{best}}$	v2016/270	4073	1.289	0.832	0.819	0.645	1.011
		15394	1.128	0.870	0.839	0.728	0.902
	v2013/189	4005	1.444	0.802	0.805	0.584	1.154
		15317	1.296	0.831	0.816	0.665	1.026
SS-GNN $_{\text{average}}$	v2016/270	4073	1.281 ± 0.021	0.822 ± 0.006	0.813 ± 0.004	0.649 ± 0.011	1.012 ± 0.016
		15394	1.181 ± 0.047	0.853 ± 0.012	0.833 ± 0.006	0.701 ± 0.024	0.920 ± 0.035
	v2013/189	4005	1.454 ± 0.050	0.795 ± 0.005	0.795 ± 0.010	0.578 ± 0.029	1.165 ± 0.055
		15317	1.347 ± 0.049	0.816 ± 0.012	0.808 ± 0.007	0.638 ± 0.027	1.074 ± 0.031

https://doi.org/10.1021/acsomega.3c00085
ACS Omega 2023, 8, 22496−22507

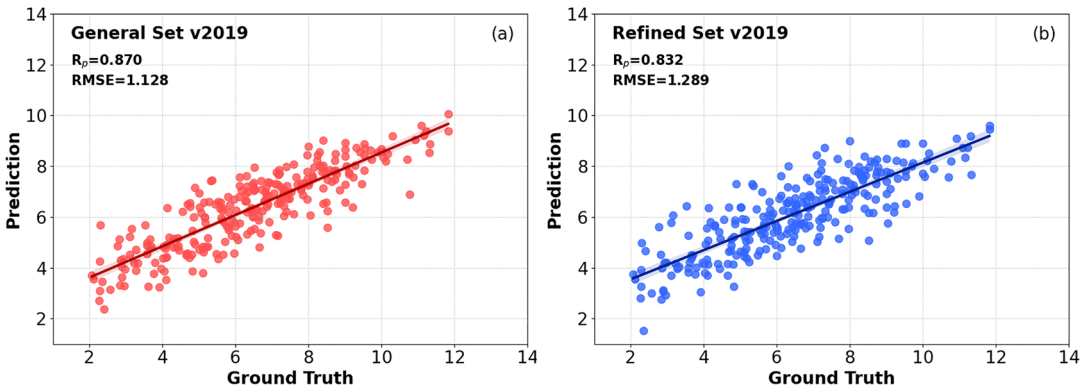


Figure 6. Correlation plot for the PDBbind v2016 core set given by the best SS-GNN models trained on (a) the general set and (b) the refined set.

Table 6. Performance Comparison Using the PDBbind v2016 Core Set and v2013 Core Set

		PDBbind v2016 core set			PDBbind v2013 core set		
architecture	training samples	test samples	RMSE (↓)	R_p (↑)	test samples	RMSE (↓)	R_p (↑)
Pafnucy	11906	290	1.420	0.780	195	1.620	0.700
SIGN	3767	290	1.316	0.797			
FAST	11717	290	1.308	0.810			
TNET-BP	3767	290	1.340	0.810			
IGN	8298	262	1.291[b]	0.811[b]			
ΔVinaRF$_{20}$	3336	285		0.816	195		0.686
OnionNet	11906	290	1.278	0.816	108	1.503	0.782
K$_{Deep}$	3767× 24[a]	290	1.270	0.820			
HPC/HWPC	3772	285	1.307	0.831	195/2764	1.483	0.784
AGL-Score	3772	285	1.271	0.833	195/3516		0.792
MP-GNN	4057	285	0.828	0.836	195/2959	0.801	0.805
PerSpect ML	3772	285	1.724	0.840	195/2764	1.956	0.793
DCML	3772	285	1.255	0.843	195/2764	1.432	0.799
Mol-PSI	4057	285	1.278	0.844	195/2959	1.383	0.821
TopBP	3767	290	1.650	**0.861**	195/2764	1.950	0.808
SS − GNN$_{refined\ set}$	4073	270	1.281	0.822	189/4005	1.454	0.795
SS − GNN$_{general\ set}$	15394	270	**1.181**	**0.853**	189/15317	**1.347**	**0.816**

[a]The data sets of K$_{Deep}$ were augmented 24 times by rotation. [b]The results of IGN are the indicators of the average model.

Table 7. Comparison of GNN-Based Models on the PDBbind v2016 Core Set

model	molecular representation	main algorithm	special features	R_p	RMSE
SIGN	complex interaction graph	polar-inspired graph attention	integrate both distance and angle information	0.797	1.316
FAST	3D voxel and molecular graph representation	3D-CNN and GNN	considering the covalent and noncovalent bonds	0.810	1.308
IGN	three molecular graphs	two independent graph convolution modules	considering the covalent bonds within protein atoms.	0.811	1.291
MP-GNN	multiphysical molecular graph representation	SAGCN, multiscale predictions with multiscale stacking	distance-related node features, featurization	0.836	**0.828**
SS-GNN	a complex graph	two-layer GIN	ignoring the covalent bonds within protein atoms.	**0.853**	1.181

4. CONCLUSION

In this paper, we have proposed a novel simple-structured graph neural network model (SS-GNN) for drug-target binding affinity (DTBA) prediction. We utilize the single undirected graph representation method based on the distance threshold to reduce the size of the complex molecular graph, thereby reducing the computational complexity of the model. The process of feature extraction and affinity prediction is straightforward. The concise graph representation and simple model architecture improve the efficiency of SS-GNN. Experiments confirm the superiority of SS-GNN, which significantly outperforms state-of-the-art methods on the PDBbind data set. However, it has not been verified which of the chemical properties we input are critical in constructing the complex graph. In addition, whether the covalent interactions between protein atoms have an effect on the interactions of the complex needs further verification.

5. EXPERIMENTAL SECTION

In this paper, all experiments are conducted according to the following procedure.

https://doi.org/10.1021/acsomega.3c00085
ACS Omega 2023, 8, 22496−22507

(1) Get the data set. The training and test data sets are downloaded from PDB. We use the v2016 and v2013 core sets as the test sets, as well as the PDBbind v2019 data set as the training set.
(2) Generate features. We generate protein−ligand complex features with the distance threshold of 5 Å, including atom features and edge features. The data with the generated features can be directly utilized for model training.
(3) Multiple hold-out validations for testing. To demonstrate the stability of the SS-GNN model, we employed multiple hold-out validations to evaluate the model.

The data sets with generated features, as well as the comprehensive experimental setups and procedures are included in the Supporting Information.

■ ASSOCIATED CONTENT

Data Availability Statement

The authors confirm that the data supporting the findings of this study are available within the Supporting Information. These data can reproduce the essential results for each experiment in the manuscript. All codes are freely accessible at https://github.com/xianyuco/SS-GNN.

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acsomega.3c00085.

Additional Tables and Figures with analyses, selection of data sets corresponding to each experiment, training and test compounds, detailed experimental information (PDF)

Data sets with generated features (ZIP)

■ AUTHOR INFORMATION

Corresponding Authors

Zhaohui Zhang − *College of Computer and Cyber Security, Hebei Normal University, Shijiazhuang 050024, China; Software College, Hebei Normal University, Shijiazhuang 050024, China;* orcid.org/0009-0001-7737-7274; Email: zhangzhaohui@hebtu.edu.cn

Shuliang Zhao − *College of Computer and Cyber Security, Hebei Normal University, Shijiazhuang 050024, China; Hebei Provincial Key Laboratory of Network and Information Security, Shijiazhuang 050024, China; Hebei Provincial Engineering Research Center for Supply Chain Big Data Analytics & Data Security, Shijiazhuang 050024, China*; Email: shuliangzhao@hebtu.edu.cn

Bo Shan − *Software College, Hebei Normal University, Shijiazhuang 050024, China; Shijiazhuang Xianyu Digital Biotechnology Co., Ltd, Shijiazhuang 050024, China*; Email: shanbo@onest.net

Authors

Shuke Zhang − *Software College, Hebei Normal University, Shijiazhuang 050024, China; Shijiazhuang Xianyu Digital Biotechnology Co., Ltd, Shijiazhuang 050024, China*

Yanzhao Jin − *Software College, Hebei Normal University, Shijiazhuang 050024, China; Shijiazhuang Xianyu Digital Biotechnology Co., Ltd, Shijiazhuang 050024, China*

Tianmeng Liu − *Software College, Hebei Normal University, Shijiazhuang 050024, China; Shijiazhuang Xianyu Digital Biotechnology Co., Ltd, Shijiazhuang 050024, China*

Qi Wang − *Software College, Hebei Normal University, Shijiazhuang 050024, China; Shijiazhuang Xianyu Digital Biotechnology Co., Ltd, Shijiazhuang 050024, China*

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.3c00085

Author Contributions

⊥S.Z. and Y.J. are equivalent authors.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This research was partially supported by the National Social Science Fund of China (No. 18ZDA200), the Hebei Provincial Key Research and Development Project of China (No. 20370301D), the Key Technology Development Project of Hebei Normal University (No. L2020K01) and the Natural Science Foundation of Hebei Province (No. H2021206352).

■ REFERENCES

(1) Mullard, A. New Drugs Cost US $2.6 Billion to Develop. *Nat. Rev. Drug Discovery* **2014**, *13*, 877.
(2) Ashburn, T. T.; Thor, K. B. Drug Repositioning: Identifying and Developing New Uses for Existing Drugs. *Nat. Rev. Drug Discovery* **2004**, *3*, 673−683.
(3) Thafar, M.; Raies, A. B.; Albaradei, S.; Essack, M.; Bajic, V. B. Comparison Study of Computational Prediction Tools for Drug-Target Binding Affinities. *Front. Chem.* **2019**, *7*, 782.
(4) Chen, X.; Yan, C. C.; Zhang, X.; Zhang, X.; Dai, F.; Yin, J.; Zhang, Y. Drug−Target Interaction Prediction: Databases, Web Servers and Computational Models. *Briefings Bioinf* **2016**, *17*, 696−712.
(5) Mei, S.; Zhang, K. A Multi-Label Learning Framework for Drug Repurposing. *Pharmaceutics* **2019**, *11*, 466.
(6) Alshahrani, M.; Hoehndorf, R. Drug Repurposing through Joint Learning on Knowledge Graphs and Literature. *BioRxiv* **2018**, No. 385617, DOI: 10.1101/385617.
(7) Öztürk, H.; Ozkirimli, E.; Özgür, A. WideDTA: Prediction of Drug-Target Binding Affinity. *arXiv* **2019**, No. 1902.04166, DOI: 10.48550/arXiv.1902.04166.
(8) Guedes, I. A.; Pereira, F. S.; Dardenne, L. E. Empirical Scoring Functions for Structure-Based Virtual Screening: Applications, Critical Aspects, and Challenges. *Front. Pharmacol.* **2018**, *9*, 1089.
(9) Huang, S.-Y.; Zou, X. An Iterative Knowledge-Based Scoring Function to Predict Protein−Ligand Interactions: I. Derivation of Interaction Potentials. *J. Comput. Chem.* **2006**, *27*, 1866−1875.
(10) Liu, Y.; Xu, Z.; Yang, Z.; Chen, K.; Zhu, W. A Knowledge-Based Halogen Bonding Scoring Function for Predicting Protein-Ligand Interactions. *J. Mol. Model.* **2013**, *19*, 5015−5030.
(11) Li, J.; Fu, A.; Zhang, L. An Overview of Scoring Functions Used for Protein−Ligand Interactions in Molecular Docking. *Interdiscip. Sci.: Comput. Life Sci.* **2019**, *11*, 320−328.
(12) King, E.; Aitchison, E.; Li, H.; Luo, R. Recent Developments in Free Energy Calculations for Drug Discovery. *Front. Mol. Biosci.* **2021**, *8*, No. 712085, DOI: 10.3389/fmolb.2021.712085.
(13) Abel, R.; Wang, L.; Harder, E. D.; Berne, B.; Friesner, R. A. Advancing Drug Discovery through Enhanced Free Energy Calculations. *Acc. Chem. Res.* **2017**, *50*, 1625−1632.
(14) Yamanishi, Y.; Kotera, M.; Kanehisa, M.; Goto, S. Drug-Target Interaction Prediction from Chemical, Genomic and Pharmacological Data in an Integrated Framework. *Bioinformatics* **2010**, *26*, i246−i254.
(15) Nascimento, A. C.; Prudêncio, R. B.; Costa, I. G. A Multiple Kernel Learning Algorithm for Drug-Target Interaction Prediction. *BMC Bioinf* **2016**, *17*, 1−16.
(16) Cheng, Z.; Zhou, S.; Wang, Y.; Liu, H.; Guan, J.; Chen, Y.-P. P. Effectively Identifying Compound-Protein Interactions by Learning

from Positive and Unlabeled Examples. *IEEE/ACM Trans. Comput. Biol. Bioinf.* **2016**, *15*, 1832−1843.
(17) Ballester, P. J.; Schreyer, A.; Blundell, T. L. Does a More Precise Chemical Description of Protein−Ligand Complexes Lead to More Accurate Prediction of Binding Affinity? *J. Chem. Inf. Model.* **2014**, *54*, 944−955.
(18) Li, H.; Leung, K.-S.; Wong, M.-H.; Ballester, P. J. Improving AutoDock Vina Using Random Forest: The Growing Accuracy of Binding Affinity Prediction by the Effective Exploitation of Larger Data Sets. *Mol. Inf.* **2015**, *34*, 115−126.
(19) Wang, C.; Zhang, Y. Improving Scoring-Docking-Screening Powers of Protein−Ligand Scoring Functions Using Random Forest. *J. Comput. Chem.* **2017**, *38*, 169−177.
(20) Durrant, J. D.; McCammon, J. A. NNScore: A Neural-Network-Based Scoring Function for the Characterization of Protein− Ligand Complexes. *J. Chem. Inf. Model.* **2010**, *50*, 1865−1871.
(21) Fujimoto, K. J.; Minami, S.; Yanai, T. Machine-Learning-and Knowledge-Based Scoring Functions Incorporating Ligand and Protein Fingerprints. *ACS Omega* **2022**, *7*, 19030−19039.
(22) Sánchez-Cruz, N.; Medina-Franco, J. L.; Mestres, J.; Barril, X. Extended Connectivity Interaction Features: Improving Binding Affinity Prediction through Chemical Description. *Bioinformatics* **2021**, *37*, 1376−1382.
(23) Nguyen, D. D.; Wei, G.-W. AGL-Score: Algebraic Graph Learning Score for Protein−Ligand Binding Scoring, Ranking, Docking, and Screening. *J. Chem. Inf. Model.* **2019**, *59*, 3291−3304.
(24) Liu, X.; Feng, H.; Wu, J.; Xia, K. Dowker Complex Based Machine Learning (DCML) Models for Protein−Ligand Binding Affinity Prediction. *PLoS Comput. Biol.* **2022**, *18*, No. e1009943.
(25) Liu, X.; Wang, X.; Wu, J.; Xia, K. Hypergraph-Based Persistent Cohomology (HPC) for Molecular Representations in Drug Design. *Briefings Bioinf* **2021**, *22*, bbaa411.
(26) Meng, Z.; Xia, K. Persistent Spectral−Based Machine Learning (PerSpect ML) for Protein−Ligand Binding Affinity Prediction. *Sci. Adv.* **2021**, *7*, No. eabc5329.
(27) Cang, Z.; Mu, L.; Wei, G.-W. Representability of Algebraic Topology for Biomolecules in Machine Learning Based Scoring and Virtual Screening. *PLoS Comput. Biol.* **2018**, *14*, No. e1005929.
(28) Karimi, M.; Wu, D.; Wang, Z.; Shen, Y. DeepAffinity: Interpretable Deep Learning of Compound−Protein Affinity through Unified Recurrent and Convolutional Neural Networks. *Bioinformatics* **2019**, *35*, 3329−3338.
(29) Öztürk, H.; Özgür, A.; Ozkirimli, E. DeepDTA: Deep Drug−Target Binding Affinity Prediction. *Bioinformatics* **2018**, *34*, i821−i829.
(30) Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints. *J. Chem. Inf. Model.* **2010**, *50*, 742−754.
(31) Liu, K.; Sun, X.; Jia, L.; Ma, J.; Xing, H.; Wu, J.; Gao, H.; Sun, Y.; Boulnois, F.; Fan, J. Chemi-Net: A Molecular Graph Convolutional Network for Accurate Drug Property Prediction. *Int. J. Mol. Sci.* **2019**, *20*, 3389.
(32) Lim, J.; Ryu, S.; Park, K.; Choe, Y. J.; Ham, J.; Kim, W. Y. Predicting Drug−Target Interaction Using a Novel Graph Neural Network with 3D Structure-Embedded Graph Representation. *J. Chem. Inf. Model.* **2019**, *59*, 3981−3988.
(33) Hassan-Harrirou, H.; Zhang, C.; Lemmin, T. RosENet: Improving Binding Affinity Prediction by Leveraging Molecular Mechanics Energies with an Ensemble of 3D Convolutional Neural Networks. *J. Chem. Inf. Model.* **2020**, *60*, 2791−2802.
(34) Ragoza, M.; Hochuli, J.; Idrobo, E.; Sunseri, J.; Koes, D. R. Protein−Ligand Scoring with Convolutional Neural Networks. *J. Chem. Inf. Model.* **2017**, *57*, 942−957.
(35) Stepniewska-Dziubinska, M. M.; Zielenkiewicz, P.; Siedlecki, P. Development and Evaluation of a Deep Learning Model for Protein−Ligand Binding Affinity Prediction. *Bioinformatics* **2018**, *34*, 3666−3674.
(36) Wallach, I.; Dzamba, M.; Heifets, A. AtomNet: A Deep Convolutional Neural Network for Bioactivity Prediction in Structure-Based Drug Discovery. *arXiv* **2015**, No. 1510.02855, DOI: 10.48550/arXiv.1510.02855.
(37) Zheng, L.; Fan, J.; Mu, Y. Onionnet: A Multiple-Layer Intermolecular-Contact-Based Convolutional Neural Network for Protein−Ligand Binding Affinity Prediction. *ACS omega* **2019**, *4*, 15956−15965.
(38) Wang, Z.; Zheng, L.; Liu, Y.; Qu, Y.; Li, Y.-Q.; Zhao, M.; Mu, Y.; Li, W. OnionNet-2: A Convolutional Neural Network Model for Predicting Protein−Ligand Binding Affinity Based on Residue-Atom Contacting Shells. *Front. Chem.* **2021**, 913.
(39) Jiménez, J.; Skalic, M.; Martinez-Rosell, G.; De Fabritiis, G. K. deep: Protein−Ligand Absolute Binding Affinity Prediction Via 3d-Convolutional Neural Networks. *J. Chem. Inf. Model.* **2018**, *58*, 287−296.
(40) Jones, D.; Kim, H.; Zhang, X.; Zemla, A.; Stevenson, G.; Bennett, W. D.; Kirshner, D.; Wong, S. E.; Lightstone, F. C.; Allen, J. E. Improved Protein−Ligand Binding Affinity Prediction with Structure-Based Deep Fusion Inference. *J. Chem. Inf. Model.* **2021**, *61*, 1583−1592.
(41) Nguyen, D. D.; Cang, Z.; Wu, K.; Wang, M.; Cao, Y.; Wei, G.-W. Mathematical Deep Learning for Pose and Binding Affinity Prediction and Ranking in D3R Grand Challenges. *J. Comput.-Aided Mol. Des.* **2019**, *33*, 71−82.
(42) Nguyen, D. D.; Cang, Z.; Wei, G.-W. A Review of Mathematical Representations of Biomolecular Data. *Phys. Chem. Chem. Phys.* **2020**, *22*, 4343−4367.
(43) Cang, Z.; Wei, G.-W. TopologyNet: Topology Based Deep Convolutional and Multi-Task Neural Networks for Biomolecular Property Predictions. *PLoS Comput. Biol.* **2017**, *13*, No. e1005690.
(44) Jiang, P.; Chi, Y.; Li, X.-S.; Meng, Z.; Liu, X.; Hua, X.-S.; Xia, K. Molecular Persistent Spectral Image (Mol-PSI) Representation for Machine Learning Models in Drug Design. *Briefings Bioinf* **2022**, *23*, bbab527.
(45) Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Yu, P. S. A Comprehensive Survey on Graph Neural Networks. *IEEE transactions on neural networks and learning systems* **2021**, *32*, 4−24.
(46) Nguyen, T.; Le, H.; Quinn, T. P.; Nguyen, T.; Le, T. D.; Venkatesh, S. GraphDTA: Predicting Drug−Target Binding Affinity with Graph Neural Networks. *Bioinformatics* **2021**, *37*, 1140−1147.
(47) Niepert, M.; Ahmed, M.; Kutzkov, K. Learning Convolutional Neural Networks for Graphs. *Proc 33rd Int. Conf. Mach. Learn.* **2016**, 2014−2023.
(48) Gao, H.; Wang, Z.; Ji, S. Large-Scale Learnable Graph Convolutional Networks. *Proc. 24th ACM SIGKDD Int. Conf. Know. Discover. Data Mining* **2018**, 1416−1424.
(49) Sun, M.; Zhao, S.; Gilvary, C.; Elemento, O.; Zhou, J.; Wang, F. Graph Convolutional Networks for Computational Drug Development and Discovery. *Briefings Bioinf* **2020**, *21*, 919−935.
(50) Zhou, J.; Li, S.; Huang, L.; Xiong, H.; Wang, F.; Xu, T.; Xiong, H.; Dou, D. Distance-Aware Molecule Graph Attention Network for Drug-Target Binding Affinity Prediction. *arXiv* **2020**, No. 2012.09624, DOI: 10.48550/arXiv.2012.09624.
(51) Gasteiger, J.; Groß, J.; Günnemann, S. Directional Message Passing for Molecular Graphs. *8th International Conference on Learning Representations*, 2020.
(52) Danel, T.; Spurek, P.; Tabor, J.; Śmieja, M.; Struski, Ł.; Słowik, A.; Maziarka, Ł. Spatial Graph Convolutional Networks. *International Conference on Neural Information Processing* **2020**, *1333*, 668−675.
(53) Li, S.; Zhou, J.; Xu, T.; Huang, L.; Wang, F.; Xiong, H.; Huang, W.; Dou, D.; Xiong, H. Structure-Aware Interactive Graph Neural Networks for the Prediction of Protein−Ligand Binding Affinity. *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining* **2021**, 975−985.
(54) Jiang, D.; Hsieh, C.-Y.; Wu, Z.; Kang, Y.; Wang, J.; Wang, E.; Liao, B.; Shen, C.; Xu, L.; Wu, J.; et al. InteractionGraphNet: A Novel and Efficient Deep Graph Representation Learning Framework for Accurate Protein−Ligand Interaction Predictions. *J. Med. Chem.* **2021**, *64*, 18209−18232.

https://doi.org/10.1021/acsomega.3c00085
ACS Omega 2023, 8, 22496−22507

(55) Li, X.-S.; Liu, X.; Lu, L.; Hua, X.-S.; Chi, Y.; Xia, K. Multiphysical Graph Neural Network (MP-GNN) for COVID-19 Drug Design. *Briefings Bioinf* **2022**, *23*, bbac231.
(56) Karlov, D. S.; Sosnin, S.; Fedorov, M. V.; Popov, P. GraphDelta: MPNN Scoring Function for the Affinity Prediction of Protein−Ligand Complexes. *ACS omega* **2020**, *5*, 5150−5159.
(57) Szocinski, T.; Nguyen, D. D.; Wei, G.-W. AweGNN: Auto-Parametrized Weighted Element-Specific Graph Neural Networks for Molecules. *Comput. Biol. Med.* **2021**, *134*, 104460.
(58) Moon, S.; Zhung, W.; Yang, S.; Lim, J.; Kim, W. Y. PIGNet: A Physics-Informed Deep Learning Model Toward Generalized Drug−Target Interaction Predictions. *Chem. Sci.* **2022**, *13*, 3661−3673.
(59) Yang, Z.; Zhong, W.; Zhao, L.; Chen, C. Y.-C. Mgraphdta: Deep Multiscale Graph Neural Network for Explainable Drug−Target Binding Affinity Prediction. *Chem. Sci.* **2022**, *13*, 816−833.
(60) Yang, Z.; Zhong, W.; Lv, Q.; Dong, T.; Yu-Chian Chen, C. Geometric Interaction Graph Neural Network for Predicting Protein−Ligand Binding Affinities from 3D Structures (GIGN). *J. Phys. Chem. Lett.* **2023**, *14*, 2020−2033.
(61) Xu, K.; Hu, W.; Leskovec, J.; Jegelka, S. How Powerful are Graph Neural Networks? International Conference on Learning Representations, New Orleans, May 6-9, 2019.
(62) Jiang, D.; Wu, Z.; Hsieh, C.-Y.; Chen, G.; Liao, B.; Wang, Z.; Shen, C.; Cao, D.; Wu, J.; Hou, T. Could Graph Neural Networks Learn Better Molecular Representation for Drug Discovery? A Comparison Study of Descriptor-Based and Graph-Based Models. *J. Cheminf.* **2021**, *13*, 1−23.
(63) Wieder, O.; Kohlbacher, S.; Kuenemann, M.; Garon, A.; Ducrot, P.; Seidel, T.; Langer, T. A Compact Review of Molecular Property Prediction with Graph Neural Networks. *Drug Discovery Today: Technol.* **2020**, *37*, 1−12.
(64) Febrinanto, F. G.; Xia, F.; Moore, K.; Thapa, C.; Aggarwal, C. Graph Lifelong Learning: A Survey. *IEEE Computational Intelligence Magazine* **2023**, *18*, 32−51.
(65) Han, J.; Rong, Y.; Xu, T.; Huang, W. Geometrically Equivariant Graph Neural Networks: A Survey. *arXiv* **2022**, No. 2202.07230.v3, DOI: 10.48550/arXiv.2202.07230.
(66) Zhou, J.; Li, S.; Huang, L.; Xiong, H.; Wang, F.; Xu, T.; Xiong, H.; Dou, D.; et al. Graph Attention Networks. *Stat* **2017**, *1050*, 10−48550.
(67) Xu, K.; Li, C.; Tian, Y.; Sonobe, T.; Kawarabayashi, K.-i.; Jegelka, S. Representation Learning on Graphs with Jumping Knowledge Networks. *Proceedings of the 35th International conference on machine learning*; 2018; pp 5453−5462.
(68) Hamilton, W.; Ying, Z.; Leskovec, J. Inductive Representation Learning on Large Graphs. *Advances in neural information processing systems*; 2017; Vol.: *30*.
(69) Murphy, R. L.; Srinivasan, B.; Rao, V.; Ribeiro, B. Janossy Pooling: Learning Deep Permutation-Invariant Functions for Variable-Size Inputs. International Conference on Learning Representations, New Orleans, May 6-9, 2019.
(70) Grattarola, D.; Zambon, D.; Bianchi, F. M.; Alippi, C. Understanding Pooling in Graph Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems* **2022**, 1−11.
(71) Lee, J.; Lee, I.; Kang, J. Self-Attention Graph Pooling. *International conference on machine learning*; 2019; pp 3734−3743.
(72) Wang, R.; Fang, X.; Lu, Y.; Wang, S. The PDBbind Database: Collection of Binding Affinities for Protein−Ligand Complexes with Known Three-Dimensional Structures. *J. Med. Chem.* **2004**, *47*, 2977−2980.
(73) Wang, R.; Fang, X.; Lu, Y.; Yang, C.-Y.; Wang, S. The PDBbind Database: Methodologies and Updates. *J. Med. Chem.* **2005**, *48*, 4111−4119.
(74) Su, M.; Yang, Q.; Du, Y.; Feng, G.; Liu, Z.; Li, Y.; Wang, R. Comparative Assessment of Scoring Functions: The CASF-2016 Update. *J. Chem. Inf. Model.* **2019**, *59*, 895−913.
(75) Li, Y.; Han, L.; Liu, Z.; Wang, R. Comparative Assessment of Scoring Functions on an Updated Benchmark: 2. Evaluation Methods and General Results. *J. Chem. Inf. Model.* **2014**, *54*, 1717−1736.
(76) Wackerly, D.; Mendenhall, W.; Scheaffer, R. L. *Mathematical Statistics with Applications*; Cengage Learning, 2014.
(77) Gönen, M.; Heller, G. Concordance Probability and Discriminatory Power in Proportional Hazards Regression. *Biometrika* **2005**, *92*, 965−970.

https://doi.org/10.1021/acsomega.3c00085
ACS Omega 2023, 8, 22496−22507