OXFORD

Briefings in Bioinformatics, 2022, **23(4)**, 1–10
https://doi.org/10.1093/bib/bbac231
Advance access publication date: 14 June 2022
Problem Solving Protocol

Multiphysical graph neural network (MP-GNN) for COVID-19 drug design

Xiao-Shuang Li, Xiang Liu, Le Lu, Xian-Sheng Hua, Ying Chi and Kelin Xia
Corresponding author: Kelin Xia, xiakelin@ntu.edu.sg

Abstract

Graph neural networks (GNNs) are the most promising deep learning models that can revolutionize non-Euclidean data analysis. However, their full potential is severely curtailed by poorly represented molecular graphs and features. Here, we propose a multiphysical graph neural network (MP-GNN) model based on the developed multiphysical molecular graph representation and featurization. All kinds of molecular interactions, between different atom types and at different scales, are systematically represented by a series of scale-specific and element-specific graphs with distance-related node features. From these graphs, graph convolution network (GCN) models are constructed with specially designed weight-sharing architectures. Base learners are constructed from GCN models from different elements at different scales, and further consolidated together using both one-scale and multi-scale ensemble learning schemes. Our MP-GNN has two distinct properties. First, our MP-GNN incorporates multiscale interactions using more than one molecular graph. Atomic interactions from various different scales are not modeled by one specific graph (as in traditional GNNs), instead they are represented by a series of graphs at different scales. Second, it is free from the complicated feature generation process as in conventional GNN methods. In our MP-GNN, various atom interactions are embedded into element-specific graph representations with only distance-related node features. A unique GNN architecture is designed to incorporate all the information into a consolidated model. Our MP-GNN has been extensively validated on the widely used benchmark test datasets from PDBbind, including PDBbind-v2007, PDBbind-v2013 and PDBbind-v2016. Our model can outperform all existing models as far as we know. Further, our MP-GNN is used in coronavirus disease 2019 drug design. Based on a dataset with 185 complexes of inhibitors for severe acute respiratory syndrome coronavirus (SARS-CoV/SARS-CoV-2), we evaluate their binding affinities using our MP-GNN. It has been found that our MP-GNN is of high accuracy. This demonstrates the great potential of our MP-GNN for the screening of potential drugs for SARS-CoV-2. **Availability:** The Multiphysical graph neural network (MP-GNN) model can be found in https://github.com/Alibaba-DAMO-DrugAI/MGNN. Additional data or code will be available upon reasonable request.

Keywords: Graph neural network, Graph representation and featurization, Protein–ligand binding, Drug design, Ensemble learning

Introduction

So far, more than 262 million infections and 5 million fatalities haves been succumbed to the new severe acute respiratory syndrome coronavirus (SARS-CoV-2) in the coronavirus disease 2019 (COVID-19) pandemic which has swept across all 213 countries and territories. The significance of designing efficient antibodies and drugs for COVID-19 cannot be overemphasized. Artificial intelligence-based models have demonstrated great power in various steps in drug design [1]. Among these models are graph neural network (GNN) models, which are end-to-end learning models that take in a molecular graph representation and directly output the prediction. Originally, GNNs were developed for the analysis of large-scale network data with the main focus of predicting the properties of new nodes or edges within the network. Recently, GNNs have been used in biomolecular data analysis and achieved great performance for various steps in drug design and discovery [2–10]. Among these models, AquaSol [2] uses directed acyclic graph based recursive neural networks to predict molecular solubility. In DeepVS [3], an effective atom context representation is employed that can take into consideration protein–ligand complex properties. An integrated model of the compound-structure-based GNN and the protein-sequence-based convolution neural network (CNN) is developed for compound protein interactions [7]. GAN model is introduced for chemical stability prediction

Xiao-Shuang Li Xiao-Shuang Li is a PhD student in Shanghai Jiao Tong University, and also a research intern at the Alibaba DAMO Academy.
Xiang Liu Xiang Liu is a PhD student from Nankai University in China. He is a visiting student in Nanyang Technological University from December 2019 to June 2020.
Le Lu Le Lu is IEEE Fellow. He is the Head of Medical AI research and development of Alibaba Group, and also Senior Director of DAMO Academy USA.
Xian-Sheng Hua Xian-Sheng Hua is IEEE Fellow. He is the head of CityBrain Lab and leads the Artitificial Intelligence Center of DAMO Academy in Alibaba Group.
Ying Chi Ying Chi is the team leader of Drug Discovery Intelligence at the Alibaba DAMO Academy. She did PhD in Imperial College London and Postdoctoral research in Oxford University in UK. Her current research and development interest is all types of AI methods for various drug discovery problems, e.g. virtual screening, protein and immunity related.
Kelin Xia Kelin Xia is an assistant professor at School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore. His research interests are topological data analysis, molecular-based mathematical biology and machine learning.
Received: March 7, 2022. **Revised:** April 24, 2022. **Accepted:** May 18, 2022

in DeepChemStable [8]. A convolution spatial graph embedding layer (C-SGEL) based graph convolution network (GCN) model is developed for molecular property prediction [9]. A structure-aware interactive GNN is designed to learn essential long-range interactions among atoms and to fully utilize biomolecular structural information [11]. GNN models have also been used in drug–target affinity prediction [11–15], antibiotic discovery [16], protein–protein binding affinity change upon mutation [17] and various other drug discovery and development [18].

Even though GNNs have shown great promise for drug design, their full potential has been hindered by the inefficient graph topological representations and featurization. Currently, most biomolecular GNNs use the covalent-bond-based graph representation, which is to model a molecule as a graph with atoms represented as nodes and covalent bonds as edges. Node and edge features are then generated from different types of physical, chemical and biological properties. However, these covalent-bond-based molecular topologies fail to efficiently characterize non-covalent interactions, which can be of great importance for biomolecular complexes, including protein–protein complexes, protein–ligand complexes, protein–DNA/RAN complexes and DNA/RNA–ligand complexes. To alleviate the problem, a fixed cutoff-distance-based molecular graph representation has been developed. However, molecular interactions are usually of different scales. The fixed cutoff-distance-based topology tends to miss a great amount of information and it is nontrivial to identify the 'best' cutoff distance. Currently, the bottleneck for the design of efficient molecular GNN models is the suitable topological representations and featurization that characterize the multiphysical properties of biomolecules.

Here, we develop multiphysical molecular graph representations and featurization. Based on them, we propose a multiphysical graph neural network (MP-GNN) model. Our MP-GNN employs an ensemble learning scheme to incorporate both scale-specific GNN models and element-specific GNN models. It has been found that our MP-GNN model can deliver state-of-the-art results for protein–ligand binding affinity prediction and achieve extremely high accuracy in SARS-CoV BA dataset, which contains 185 M^{pro}-ligand complexes and their experimental binding affinities.

Results

Physically, atomic interactions within and between molecules are of various types, ranging from strong ones such as covalent bonds, disulfide bonds, ionic bonds, hydrogen bonds, to relatively weaker ones, such as van der Waals forces, electrostatic interactions, hydrophobic and hydrophilic effects. Mathematically, the atomic interaction between two atoms with coordinates r_i and r_j can be defined as an interaction function $\Phi(||\mathbf{r}_i - \mathbf{r}_j||)$ with $\|\mathbf{r}_i - \mathbf{r}_j\|$ the Euclidean distance. To model the multiscale effects, the scale (or resolution) related kernel functions are used. Among them, the most common ones are the generalized exponential kernels and the generalized Lorentz kernels. For two atoms r_i and r_j, their atomic interaction can be modeled by the generalized exponential kernel as follows:

$$\Phi(||\mathbf{r}_i - \mathbf{r}_j||; \eta) = e^{-(||\mathbf{r}_i - \mathbf{r}_j||/\eta)^\kappa}, \tag{1}$$

or by generalized Lorentz kernel as

$$\Phi(||\mathbf{r}_i - \mathbf{r}_j||; \eta) = \frac{1}{1 + (||\mathbf{r}_i - \mathbf{r}_j||/\eta)^\kappa}. \tag{2}$$

Here, η is scale (or resolution) parameter, and κ is order parameter, which is usually taken as 2. Based on rigidity–flexibility model, we can define the node importance using rigidity index as follows:

$$\mu(\mathbf{r}_i; \eta) = \sum_j w_j \Phi(||\mathbf{r}_i - \mathbf{r}_j||; \eta), \tag{3}$$

where w_j is an atomic type-dependent weight. Note that kernel functions with different scale values will focus on atomic interactions at different scales. If a small η value is used, the kernels characterize only strong covalent interactions with the values for other interactions at longer distance as (nearly) 0. In contrast, under a larger η value, relatively weaker interactions will also be included. Node importance will vary with scale values in a similar way.

In our scale-specific graph representations, molecules are modeled by a series of graphs systematically generated from different scales. Mathematically, a fully connected molecular graph is generated with scale-related weight value, i.e. the atomic interaction from Eq.(1) or Eq.(2), on each edge. Based on the scale-specific graph representation, the normalized adjacent matrix can be defined as

$$\hat{A}(i,j) = \begin{cases} \Phi(||\mathbf{r}_i - \mathbf{r}_j||; \eta), & i \neq j \\ 0, & i = j, \end{cases} \tag{4}$$

and the normalized degree matrix can be defined as

$$\hat{D}(i,j) = \begin{cases} \mu(\mathbf{r}_i; \eta), & i = j \\ 0, & i \neq j. \end{cases} \tag{5}$$

In this way, the scale effects are incorporated into molecular graph representation.

Further, we propose a new type of node feature vector that is solely dependent on atomic interaction function Φ. For the i-th node, an n-th dimensional node feature vector $\mathbf{v}^i(\eta) = (v_1^i(\eta), v_2^i(\eta), ..., v_n^i(\eta))$ is defined as follows:

$$v_k^i(\eta) = \sum_{j=1} \chi(x_{k-1} \leq \Phi(||\mathbf{r}_i - \mathbf{r}_j||; \eta) < x_k), k = 1, 2.., n. \tag{6}$$

Here, we assume all atomic interactions $\Phi(||\mathbf{r}_i - \mathbf{r}_j||; \eta)$ are within the region $[0, x_{max}]$, which is equally divided into n intervals $\{(x_{k-1}, x_k); k = 1, 2, .., n\}$ with $x_0 = 0$ and $x_n = x_{max}$. The indicator function χ equals to 1 if the following condition is satisfied and 0 otherwise. Mathematically, the node vector is the frequencies (or the numbers) of atomic interactions within a certain range.

Finally, molecules are usually of various sizes, which will result in different-sized molecular graphs. To facilitate weight-sharing among different molecular graphs, we use node importance μ as in Eq. (3) to remove less important extra nodes, so that a same-sized molecular graph is obtained.

Element-specific graph for GNN

Other than scale effects, element types are the other key factor for multiphysical atomic interactions. For instance, carbon atoms are usually associated with hydrophobic interactions, while nitrogen and oxygen atoms are correlated to hydrophilic interactions and/or hydrogen bonds. To enable a systematic description of atomic interactions, we consider element-specific graph representations [19]. Recently, the combination of element-specific representations and machine learning models has achieved great success in drug design [19–30]. More recently, an element-specific GNN model has been developed and has achieved state-of-the-art performance in quantitative toxicity analysis and solvation prediction [10].

The essential idea for element-specific representations is to decompose a molecule into a series of atom-sets, which composed of certain specific types of elements. In general, a protein molecule is composed of roughly five most important elements, denoted as $\mathbb{E}_P$ = [C, N, O, S, H]. A DNA or RNA also have five most important elements, denoted as $\mathbb{E}_D$ = [C, N, O, P, H]. For ligands or chemical molecules, they tend to have more types of elements. Here, we consider only nine types of most-commonly used ones, and denote as $\mathbb{E}_L$ = [C, N, O, S, H, F, Cl, Br, I]. In general, an element-specific GNN model contains a series of molecular graphs that are constructed based on different element types. For instance, a protein can be represented by a series of element-specific graphs, including single-element graphs (C-graph, N-graph, O-graph, S-graph and H-graph), double-element graphs (CN-graph, CO-graph, CS-graph, CH-graph, NO-graph, NS-graph, NH-graph, OS-graph, OH-graph and SH-graph), three-element graphs and other graphs with more types of elements. Each element-specific graph characterizes certain type of atomic interactions. Note that the all-atom graph as in previous GNN models is just a special case of element-specific graph. Normally, we do not need to use all the combinations [25, 29, 30]. To balance the computational cost and model accuracy, we usually only consider the element-specific graphs with sufficient amount of atoms. For instance, ligand molecules may contain Cl atom but they usually have only one or two Cl atoms. A Cl-graph will be meaningless. However, the Cl atom can be important for ligand properties. So we can consider multiple-element graphs, such as CNCl-graph, COCl-graph, etc.

Multiphysical graph neural network

In our MP-GNN model, a series of scale-specific and element-specific graphs are generated from molecules. From each graph, a GNN architecture is constructed. To significantly reduce the learning parameters, weight-sharing schemes and ensemble learning models are considered. Molecular structural topologies are of great importance for their functions. Various quantitative structure–activity/property relationship (QSAR/QSPR) have been developed to establish relations between molecular groups, motifs, conserved regions, domains and other molecular topologies with their functions [31–33]. In GNN models, weight-sharing schemes are used to characterize common molecular topologies, as similar structure topologies, defined by the same weights in GNNs, tend to induce similar functions. Moreover, weight-sharing schemes can significantly reduce parameters and network complexities. In our MP-GNN, we use the same weight schemes among the same scale-specific and element-specific graph. We also allow to use same weight schemes among relatively similar element-specific graphs, to reduce computational cost and when there is relatively less training data. Ensemble learning models use multiple base learning algorithms to boost the performance of the prediction. Here, we consider two types of ensemble learning, i.e. single-scale (one-scale) stacking and multiscale stacking. The one-scale stacking ensemble model is used to alleviate the impact of randomness caused by initialization. The multiscale stacking is for boosting the performance by the consolidation of base learners that focus on different scales and have less overlap.

MP-GNN for COVID-19 drug design

MP-GNN for protein–ligand interactions

Recently, a series of topological models have been developed for the characterization of protein–ligand interactions and have achieved great successes [25, 29, 30]. The essential idea of these models is to define special matrices that focus on interactions between the protein and the ligand, instead of interactions within either the protein or the ligand, and to construct molecular topological models based on these matrices [25].

Mathematically, we can set the protein–ligand interaction matrix M as follows:

$$M(m_i, m_j) = \begin{cases} \Phi(\|\mathbf{r}_i - \mathbf{r}_j\|; \eta), & \text{if } \mathbf{r}_i \in \mathbf{R}_P, \mathbf{r}_j \in \mathbf{R}_L \\ & \text{or } \mathbf{r}_i \in \mathbf{R}_L, \mathbf{r}_j \in \mathbf{R}_P \\ \infty, & \text{otherwise.} \end{cases} \quad (7)$$

Here, $\mathbf{r}_i$ and $\mathbf{r}_j$ are coordinates for the i- and j-th atoms, and m_i and m_j are their indices in the matrix. Two sets

$\mathbf{R}_P$ and $\mathbf{R}_L$ are atom coordinate sets for protein and ligand, respectively. Note that only interactions between protein atoms and ligand atoms are considered, while interactions between atoms within either the protein or the ligand are ignored by setting their distances as ∞, i.e. an infinitely large value. Other than the generalized kernel functions as in Eqs. (1) and (2), we can also define Euclidean-distance and electrostatics-based atomic interaction functions as

$$\Phi(||\mathbf{r}_i - \mathbf{r}_j||) = \frac{1}{||\mathbf{r}_i - \mathbf{r}_j||^\kappa}, \tag{8}$$

and

$$\Phi(||\mathbf{r}_i - \mathbf{r}_j||) = \frac{1}{1 + \exp\left(-\frac{cq_iq_j}{\|\mathbf{r}_i - \mathbf{r}_j\|}\right)}, \tag{9}$$

where q_i and q_j are partial charges for the i-th and j-th atoms, and parameter c is a constant value. All the three types of atomic interactions are considered in our MP-GNN.

Further, element-specific graphs are constructed only between protein atoms and ligand atoms. As stated above, a protein molecule is usually composed of roughly five important elements $\mathbb{E}_P$ = [C, N, O, S, H] and ligands composed of nine types $\mathbb{E}_L$ = [C, N, O, S, H, F, Cl, Br, I]. We generate a series of element-specific bipartite graphs in our MP-GNN. Each bipartite graph is composed of two sets of same-typed atoms with one set from the protein and the other from the ligand. Edges can be only formed between the two sets (thus the name of the bipartite graph), and are determined by interaction matrix as in Eq. (7). In general, when the multiscale kernel functions are used, a total of $36 = 4 * 9$ types of bipartite graphs are generated without the consideration of H atoms. Moreover, four different types of scale (or resolution) parameters are used, i.e. η =2, 5, 10 and 20 Å. Figure 1 illustrates the general architecture of our MP-GNN model for protein–ligand interaction analysis. More details of MP-GNN model can be found in Method.

Datasets We consider three most commonly used benchmark datasets for protein–ligand binding affinity prediction, including PDBbind-v2007, PDBbind-v2013 and PDBbind-v2016. All the datasets used in this paper are shown in Table 1. There are pre-train sets, training sets and test sets for the separated experiments on PDBbind-v2007, v2013 and v2016. There are intersections between the datasets, so the pre-train set is randomly selected from the non-intersected samples of one dataset. The union of these three datasets is 4413. For PDBbind v2007, the pre-train set contains 1000 items from 3114 non-intersected samples. For PDBbind v2013, 1000 from 1455 non-intersected samples and for PDBbind v2016, all 357 non-intersected complexes are used for pre-training. The core set acts as the test set for evaluation. The training set is obtained by the refined set minus the core set.

To test the performance of our model for COVID-19 drug design, we consider a SARS-CoV BA dataset, which contains 185 M^{pro}-ligand complexes and their experimental binding affinities. Among the 185 ligands, there are 44 X-ray crystal structures and the rest are in 2D SMILES strings. The software MathPose is used to predict 3D structures of those 2D ligands and generate the binding complexes of all 185 ligands with M^{pro}. To carry out the validation, we randomly split the SARS-CoV BA set into five non-overlapped folds. In each task, our MP-GNN is trained on the part of SARS-CoV BA dataset in conjunction with the PDBbind-v2019 set. More specifically, one fold (or division) is used as the validation set in each task, and the rest four folds are combined with the PDBbind-v2019 general set to form the training set. No pre-train is done before training.

Benchmark tests for MP-GNN

More than 40 different scoring functions or models have been extensively tested on the three PDBbind datasets. Figure 2 shows the comparison between our MP-GNN and the other models. The upper part depicts the overall performance, and our method is marked in red. All results are measured by Pearson correlation coefficient (denoted as R_p). Our method stays ahead of all other works for all three datasets, except second to TopBP in PDBbind-v2016. More specifically, the current best R_p on PDBbind-v2007 is 0.831 achieved by FPRC [37], while on PDBbind-v2013 and PDBbind-v2016 are 0.808 and 0.861 both achieved by TopBP [25]. Our MP-GNN surpasses the current best results on PDBbind-v2013 by 2% and stays in line with the current best results of PDBbind-v2007 with a slight advantage. On PDBbind-v2016, it is 1% lower than TopBP. In the line chart, the right part where R_p over 0.6 is dense, and the clear ranking is displayed below. It is worth mentioning that our method achieves significant improvement on the hardest dataset, PDBbind-v2013, which has a more unbalanced distribution between training and test set (See Table 1). Figure 3 demonstrates the performance for two stacking schemes and learning rate of our model on PDBbind-v2007. A more detailed illustration of our detailed results for all three datasets can be found in Tables S1 to S3.

MP-GNN for COVID-19 drug design

The COVID-19 pandemic, started in late December 2019 and caused by new SARS-CoV-2, has infected more than 262 million individuals and has caused more than 5 million fatalities in all of the continents and over 213 countries and territories by 11 November 2021. Currently, different drug targets of SARS-CoV-2, such as the main protease (M^{pro}, also called $3CL^{pro}$), papain-Like protease (PL^{pro}), RNA-dependent RNA polymerase (RdRp), 5'-to-3' helicase protein (Nsp13), have been investigated. Among them is the main protease, which is one of the best-characterized targets for coronaviruses. It has

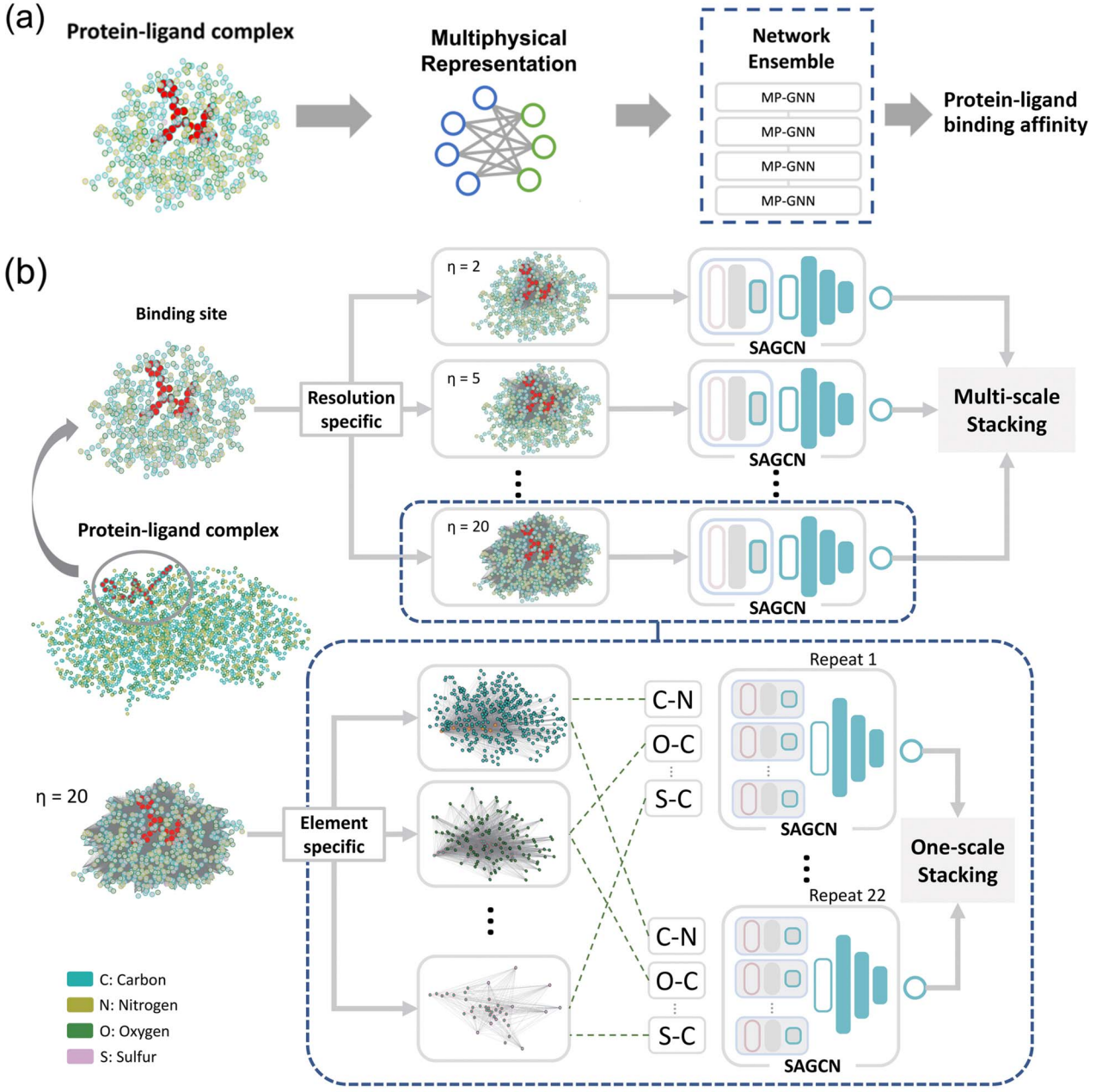

Figure 1. The framework of the protein–ligand complex binding affinity prediction system for drug design task. A complex of SARS-CoV-2 main protease inhibitor is used as an example here. This process consists of three steps: (1) generating the scale-specific graphs for the protein–ligand complex, (2) processing a group of element-specific graphs with multiphysical graph neural network for 22 repeat experiments, and performing one-scale stacking on the repeat experiments to give a prediction for one resolution and (3) giving a final decision by combining multi-scale predictions with multi-scale stacking. Nodes filled or outlined in red are from the ligand.

Table 1. A summary of our selected datasets. *mean*(*B*) refers to the mean atom number for binding sites, and *mean*(*G*) refers to the mean atom number for the un-cropped element-specific graph. The ratio between *mean*(*B*) and *mean*(*G*) describes the average complexity of the dataset

Name	Size	Pre-train size	Descriptions	*mean*(*B*)	*mean*(*G*)	$\frac{mean(G)}{mean(B)}$
PDBbind v2007 [36]	1300	1000	Refined set. Core set size 195.	583	151	0.259
PDBbind v2013 [36]	2959	1000	Refined set. Core set size 195.	195	56	0.287
PDBbind v2016 [36]	4057	358	Refined set. Core set size 285.	441	108	0.245
PDBbind v2019 [36]	17 652	–	General set.	432	114	0.264
SARS-CoV BA [34]	185	–	Inhibitors of SARS-CoV/SARS-CoV-2 main protease having experimental binding affinity.	583	149	0.256

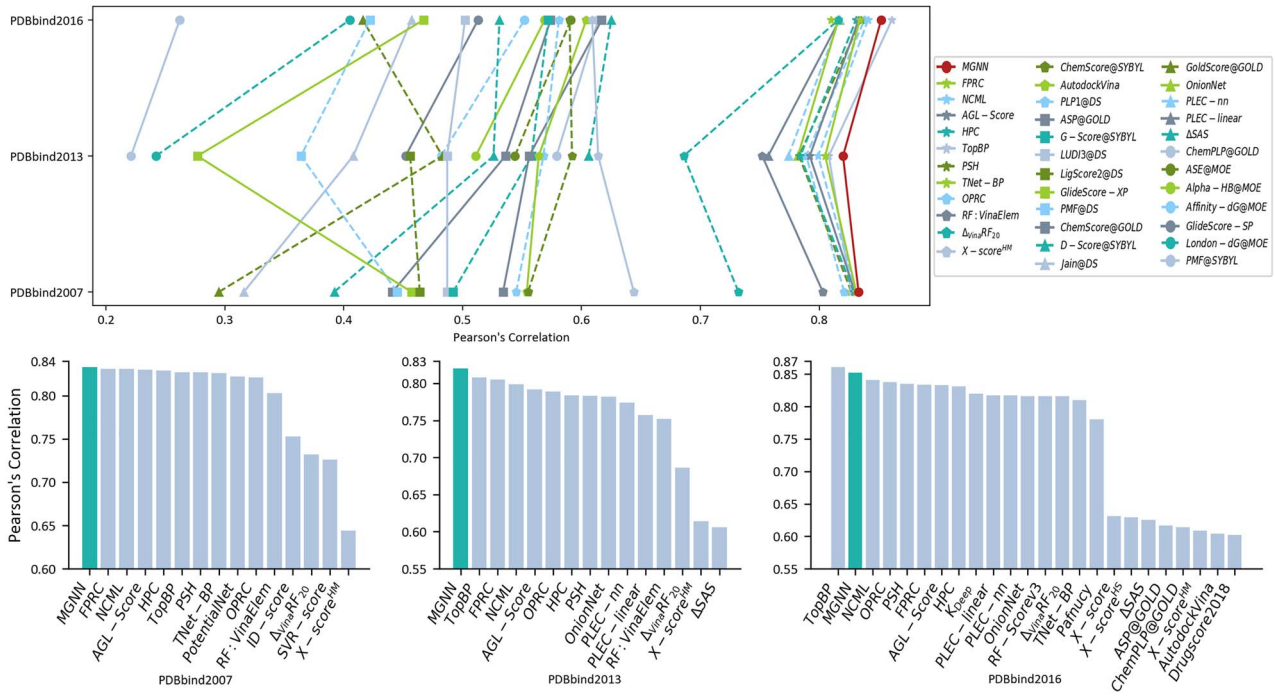

Figure 2. Comparison with recent works on three datasets, including topology-based methods, image-based methods and traditional molecular descriptor-based methods. The performances of other models are taken from [25, 34, 35]. The upper part is an overall comparison. The lower part is a clear performance ranking of works with R_p higher than 0.6 on three datasets. All results are measured with R_p.

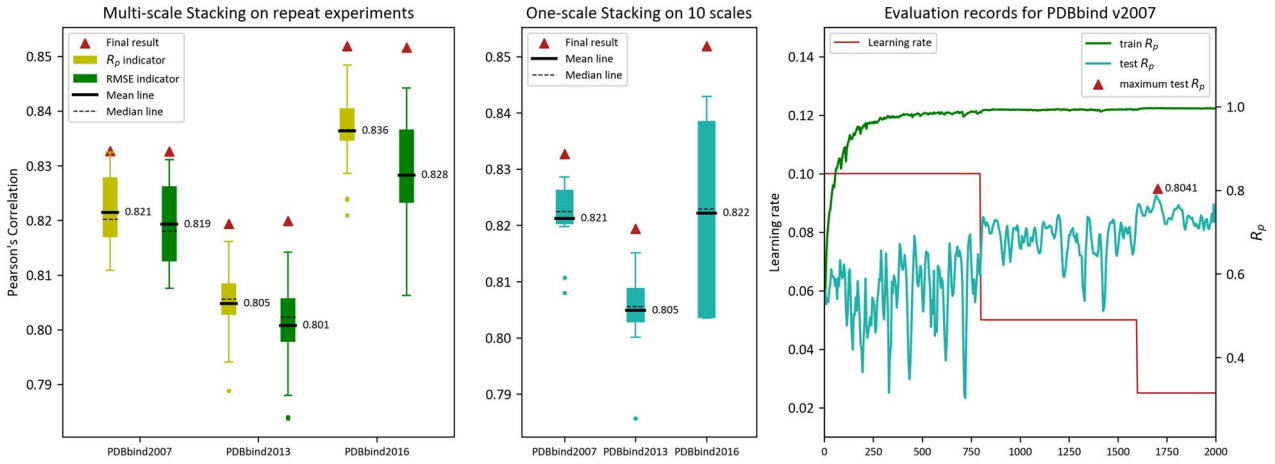

Figure 3. The left and middle box charts depict the range of performance for two stacking schemes. The line chart on the right shows the decay rule for learning rate and the R_p curves for training and test on the first repeat experiment for PDBbind-v2007 with the exponential kernel, $\eta = 10$.

been found that although the overall sequence identity between SARS-CoV and SARS-CoV-2 is just 80%, the M^{pro} of SARS-CoV-2 shares 96.08% sequence identity to that of SARS-CoV. The great gene conservation provides the opportunity for drug repurposing, i.e. use of SARS M^{pro} inhibitor for potent of SARS-CoV-2 M^{pro} inhibitor.

Recently, a dataset of 185 inhibitors of SARS-CoV/SARS-CoV-2 M^{pro}, which have experimental binding affinities, has been collected. The efficient software MathPose has been employed to predict their 3D structures, and the binding complexes between M^{pro} and these ligands, which are denoted as SARS-CoV BA. We test our MP-GNN model on this special dataset. In order to benchmark our method against MathDL [34], which is a leading approach for binding affinity prediction on SARS-CoV BA dataset, we use the same dataset partitioning scheme and cross-validation strategy [34]. The test set is divided into five partitions for 5-fold cross validation, so the test labels are used alternately for validation. It is worth mentioning that although MP-GNN and MathDL use the same dataset and random dividing scheme, the partition can be different. The average R_p and Kendall's tau (τ) for our MP-GNN model is 0.855 and 0.654, which is better than the results of MathDL, which are 0.729 and 0.540.

Discussion

Factors that impact the improvement extent of multi-scale stacking

It is revealed in supplementary that multi-scale stacking improves the R_p by 6–7% for PDBbind refined datasets, but improves it by 15–24% for the assessment on SARS-CoV BA dataset, which is a huge gap. We assume the possible reason is the diversity and the complexity of the latter training set. For every dataset, the average atom number of the binding site, *mean*(*B*) and of the sub-graph, *mean*(*G*) are recorded in Table 1. As can be seen from Tables S1, S2 and S3, there is no obvious linear relationship between prediction complexity and *mean*(*B*). Meanwhile, the ratio between *mean*(*B*) and *mean*(*G*) is given in the last column of Table 5, which is directly related to the non-empty ratio in all sub-graphs. Under the premise that most binding sites include C, N, O, S and H, a high ratio means that more element types appear in ligand. In another word, the more the non-empty sub-graphs, the more sufficient is the information, and the more the network learns from training. As a result, we presume that this ratio has negative correlation with the task complexity. Meanwhile, on the same test set, a more diversified training set can help to obtain better results. In conclusion, we believe that the first reason for such great progress on SARS-CoV BA is the message discrepancy between training and validation set. The training set is more rich in information, thus the model handles the validation set with great facility and the stacking improves more than the dataset that has consistent training and test set. The second reason is that SARS-CoV BA has a training set that is several to 10 times larger than the PDBbind refined datasets, meanwhile includes not only most data from previous year, but also four divisions of SARS-CoV BA dataset that have similar complex structure to the validation set.

Schemes for feature fusion

Through experiments, we find that channel-wise summation for node feature fusion and concatenate for sub-graph feature fusion improves the R_p as much as possible within capability. As is mentioned, symmetric operators are more suitable for nodes with a huge amount instead of concatenation. Some works [38, 39] in the field of 3D feature learning prefer channel-wise maximum. Experiments show that channel-wise maximum filters the nodes and reserves the extreme values after feature embedding. Feature visualization reveals that these extremes originate more from the inflection, depressions and contours where the features stand out. In contrast, the features that contribute more to the complex binding affinity exist more in the chemical bond force of binding site elements than in the 3D profile of the protein. The binding affinity can be viewed as the superposition of all chemical bond force in the binding site, so using the maximum operator will lose most of the information. This explains the applicability of sum operator. On the other hand, based on the premise of limited number and length of sub-graph descriptors, concatenation operator can completely deliver the features while implicitly contain the message of atom types. Notice that the features of element-specific graph do not include atom types, and that is why symmetric operators are irrational for sub-graph feature fusion.

Ablation analysis for element-specific graph and single aggregation

Result-related figures in this section are from some ablation study designed earlier. At the beginning, a single graph is used to describe the whole binding site. It is a huge bipartite structure and we have to use a large cropping size such as 130. The network was able to converge but the result is more chaotic. Early experiments have shown that converting from a single graph to element-specific graphs makes R_p increased from 0.69 to 0.749 without any stacking ensemble. This shows that seeing things from a single scale is much clearer than looking at the whole graph directly. The subsequent ensembles improve the prediction R_p again. Then, we realized that to some extent, these sub-graphs can be viewed as complete graph, that is, any two nodes in a sub-graph can obtain each other's information through one aggregation. This means the superposition of multilayer aggregation may lead to redundant and overlapping information. So we deconstructed the graph convolution layer in MP-GNN and removed the aggregation after the first layer. The best single scale R_p increased from 0.749 to 0.767 as expected, confirming the effectiveness of single aggregation.

Method

Multiphysical GNN

Graph neural network The GNN in our MP-GNN consists of two parts, i.e. a 'head' part and a 'tail' part. The 'head' part converts the node vector information from each bipartite graph to a hidden feature vector. The 'tail' part is a fully connected neural network that learns the binding affinity from the hidden feature vector.

The 'head' part contains an convolution layer followed by an encoder. In the convolution layer, node features are convoluted as in the traditional GCN model,

$$H^{l+1} = \sigma(\hat{D}^{-1/2}\hat{A}\hat{D}^{-1/2}H^{l}W^{l}), \tag{10}$$

in which $\hat{D}$ and $\hat{A}$ denote the symmetrical and normalized degree and adjacent matrices of the graph, H^{l} the node feature matrix of the *lth* layer, W the layer-specific weight matrix and $\sigma(\cdot)$ denotes the activation function. Note that the input for H^{l} is just the node features as in Eq. (6). The encoder part consists of a fully connected layer, dropout, and followed by the activation layer.

The 'tail' part also contains two parts, i.e. a feature fusion part and encoder-based prediction part. In the feature fusion part, all node feature vectors are aggregated into a single feature vector. The commonly used fuse

operations include concatenation, sum, average, maximum and minimum of all node features. Here, we use summation of node features for each element-specific graph. Then we concatenate summation vectors from all 50 element-specific graphs into a long vector. In the second part, two encoders together with a fully connected layer are used to predict the binding affinity from the concatenated vector.

Ensemble learning The ensemble learning method is to use multiple learning models simultaneously for boosting the performance. Each learning model, which is known as a base learner, can give a prediction individually. Ensemble learning improves the prediction accuracy of each base learner by assembling them together under a certain combination strategy. The commonly used combination approaches including bagging, boosting and stacking [40]. In our MP-GNN, we focus on stacking ensemble model. The essential idea is to assign a certain weight, which is to be learned, to each base learner and use the weighted results as the final prediction. More specifically, we can denote the prediction of n number of base learners as $Y_1, Y_2, ..., Y_n$, the final prediction as $Y^{stacking}$ and the ground truth value of training set as Y. The weight for each based learner is linearly related to their prediction accuracy on training set. For instance, we can use $R_p(Y_i, Y)$, which is Pearson correlation coefficient R_p between the prediction Y_i and true value Y, as the measurements for the model accuracy. The weight for the the i-th base learner is then defined as

$$W_i^{stacking} = \frac{R_p(Y_i, Y)}{\sum_{j=1}^{n} R_p(Y_j, Y)},$$

and the final prediction results are

$$Y^{stacking} = \sum_{i=1}^{n} W_i^{stacking} Y_i.$$

Other than using R_p as accuracy measurement, we have also considered RMSE in our MP-GNN models.

MP-GNN for Covid drug design

Graph representation for protein–ligand interactions Ligands usually bind to proteins, which tend to have a much larger size, at a certain special region called binding site. Computationally, the binding site is chosen as the protein region that is within a certain cutoff distance of the ligand atoms. Here, we use 10 Å in our MP-GNN model. The protein–ligand interaction matrix M in Eq. (7) is defined only on the protein binding sites instead of the entire protein domain. Three types of atomic interaction function Φ are considered, including generalized expoential/Lorentz kernel function, Euclidean distance function and electrostatic function. Under different interaction functions, different types of element-specific graph models are constructed. As stated above, proteins and ligands in general have five and nine types of atoms, that is, $\mathbb{E}_P$ = [C, N, O, S, H] and $\mathbb{E}_L$ = [C, N, O, S, H, F, Cl, Br, I]. For generalized expoential/Lorentz kernel and Euclidean distance based atomic interaction functions, we consider only $36 = 4 * 9$ types element-specific bipartite graph representation and omit the influence from hydrogen (H) atoms. For electrostatic-based interaction function, a total $50 = 5 * 10$ types of bipartite graphs are constructed. Note that for these bipartite graphs, their sizes may vary greatly between different protein–ligand complexes and between different element combinations. In our MP-GNN, node importance is defined using rigidity index as in Eq. (3). To share the weights (in GNN model) among different graphs, we choose a same cropping size, i.e. a total of 56 nodes, for all bipartite graphs. Computationally, it is found that 56 is roughly the average size of these element-specific bipartite graphs. For large-sized graphs, we will remove the nodes that have a lower node importance. For small-sized graph, pseudo-nodes are added until a common size of 56 is reached.

In our MP-GNN model, node features are only related to atomic distances. For generalized kernel based function Φ as in Eqs. (1) and (2), we set κ to be 2 and four different scale parameters are considered, that is, η=2, 5, 10 and 20 Å. For Euclidean distance based function Φ as in Eq. (8), we set $\kappa = -1$ and let Φ simply equals to the atomic distance. The function domain of Eqs. (1), Eqs. (2) and Eqs. (8) is set to be [2 Å, 30Å] with each interval of length 1 Å, and the node vector as in Eq. (6) is of size 29. For electrostatic-based function Φ as in Eq. (9), we set the domain to [0, 1] with each interval of length 0.04, and the node vector is of size 25.

MP-GNN parameter settings The encoders in MP-GNN head have the output size of 64 and 16 for every node. After node feature fusion, preliminary sub-graph features go through an encoder with output feature length 16. Then the sub-graph feature matrix with shape (M*N, 16) is concatenated into one feature vector describing the binding site, which passes through the hidden layer with 256 and 64 neurons for final regression. Every MP-GNN sub-learner is trained for 6400 epochs to obtain the optimal model with a dropout rate of 0.5 and ELUs as the activation unit. The learning rate starts from 0.1 and decays every 800 epochs, and the decay rate is 0.5. The decay scheme is depicted in Figure 3.

Performance of MP-GNN on PBDbind datasets For PBDbind datasets, a total of 10 different scale-specific GNN models are considered based on 10 atomic interaction functions, including four different exponential kernel functions, four different Lorentz kernels, a Euclidean distance based function and an electrostatic-based function. A total of 10 GNN base learners can be obtained. The stacking models are chosen based on R_p on the training set performance. Due to the high computation cost, we conducted 22 repeated experiments with random initialization. The detailed results can be found in Tables S1 to S3. The best results in every sector are marked in bold. Note that stacking with R_p is better than the ones with RMSE.

We carry out the ablation analysis for our model based on the Tables S1 to S3. First, we focus on the effectiveness of one-scale stacking. When looking at one row of each scale separately, one-scale stacking significantly improves the result on PBDbind-v2007, PBDbind-v2013 and PBDbind-v2016 by around 6%, 8% and 8%. If we apply the multi-scale stacking without the one-scale stacking, both average and best R_p are improved by 5–10% than the single scale results. In contrast, multi-scale stacking after one-scale stacking can only boost R_p by 0–3%. This indicates that the one-scale stacking can avoid the information blind spot caused by initialization. Second, we study the effectiveness of multi-scale stacking. Before one-scale stacking, the multi-scale stacking average R_p of 22 random trials is at least 6%, 7% and 6% higher than the single scale average R_ps on all three datasets. It is obvious that there is information complement between different scales. Based on the result of one-scale stacking, multi-scale still increases the R_p by almost 3%. In comparison, we have noticed that although the one-scale stacking avoids randomness and collects the information in a single scale as much as possible, it can not overrun the Best R_p of the randomly initialized multi-scale stacking.

Performance of MP-GNN on SARS-CoV BA dataset In our MP-GNN model for SARS-CoV BA dataset, only multiscale stacking is employed. This is due to the reason that the training set has incorporated in it the PBDbind-v2019 general set, which has 17652 PBD data. Similar to PBDbind datasets, the same 10 different scale-specific GNN models are considered in our MP-GNN. Since stacking with R_p gives better accuracy, we also use R_p results on training set as the weighting scheme. From Table S4, it can be seen that the multiscale stacking improves the R_p by 15–24% for SARS-CoV BA dataset.

Key Points

Our main contributions in this paper are as follows:

- We propose the first multiphysical molecular graph representation. All kinds of molecular interactions, between different atom types and at different scales, are systematically represented by a series of scale-specific and element-specific graphs with distance-related node features.
- We develop the first multiphysical graph neural network (MP-GNN) model. Our MP-GNN is free from the complicated feature generation process. A unique GNN architecture is developed in our MP-GNN to incorporate both scale-specific and element-specific graph information into a consolidated model.
- Our model has achieved the state-of-the-art results for protein–ligand binding affinity prediction. It has been found that our model can outperform all existing models, as far as we know.
- Our model is highly accurate for the prediction of complexes of inhibitors for SARS-CoV/SARS-CoV-2. Our model has great potential for COVID-19 drug design.

Code and Data Availability

The code is available at https://github.com/Alibaba-DAMO-DrugAI/MGNN. Additional data or code would be available upon reasonable request.

Author contributions statement

K.X. conceived MP-GNN model. K.X., X-S.L. and Y.C. conceived the graph neural network and ensemble learning architecture. X.L. prepared the input data. X-S.L. and Y.C. wrote up all algorithm codes and accomplished try-run. X-S.L. conducted the experiments in large scale and analyzed the results. X-S.L. refined the network architecture. K.X. validated the results according to experience. K.X. and X-S.L. wrote up the paper, all other authors reviewed the manuscript.

Supplementary data

Supplementary data are available online at https://academic.oup.com/bib.

Acknowledgments

This work was supported by Alibaba Innovative Research (AIR) Program and Alibaba-NTU Singapore Joint Research Institute grant AN-GC-2020-002, Singapore Ministry of Education Academic Research fund Tier 1 RG109/19, and Tier 2 MOE-T2EP20220-0010, MOE-T2EP20120-0013.

References

1. Zhang L, Tan J, Han D, *et al.* From machine learning to deep learning: progress in machine intelligence for rational drug discovery. *Drug Discov Today* 2017;**22**(11):1680–5.
2. Lusci A, Pollastri G, Baldi P. Deep architectures and deep learning in chemoinformatics: the prediction of aqueous solubility for drug-like molecules. *J Chem Inf Model* 2013;**53**(7):1563–75.
3. Pereira JC, Caffarena ER, Nogueira C, *et al.* Boosting docking-based virtual screening with deep learning. *J Chem Inf Model* 2016;**56**(12):2495–506.
4. Kearnes S, McCloskey K, Berndl M, *et al.* Molecular graph convolutions: moving beyond fingerprints. *J Comput Aided Mol Des* 2016;**30**(8):595–608.
5. Gomes J, Ramsundar B, Feinberg EN, *et al.* Atomic convolutional networks for predicting protein-ligand binding affinityarXiv preprint arXiv:1703.10603. 2017.
6. Feinberg EN, Sur D, Wu ZQ, *et al.* Potentialnet for molecular property prediction. *ACS central science* 2018;**4**(11):1520–30.
7. Tsubaki M, Tomii K, Sese J. Compound–protein interaction prediction with end-to-end learning of neural networks for graphs and sequences. *Bioinformatics* 2019;**35**(2):309–18.
8. Li X, Yan X, Qiong G, *et al.* Deepchemstable: chemical stability prediction with an attention-based graph convolution network. *J Chem Inf Model* 2019;**59**(3):1044–9.
9. Wang X, Li Z, Jiang M, *et al.* Molecule property prediction based on spatial graph embedding. *J Chem Inf Model* 2019;**59**(9):3817–28.
10. Szocinski T, Nguyen DD, Wei G-W. AweGNN: Auto-parametrized weighted element-specific graph neural networks for molecules. *Comput Biol Med* 2021;**134**:104460.

11. Li S, Zhou J, Tong X, *et al.* (eds). Structure-aware interactive graph neural networks for the prediction of protein-ligand binding affinity. In: *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, 975–85.
12. Nguyen T, Le H, Quinn TP, *et al.* GraphDTA: Predicting drug–target binding affinity with graph neural networks. *Bioinformatics* 2021;**37**(8):1140–7.
13. Lin X. DeepGS: Deep representation learning of graphs and sequences for drug-target binding affinity predictionarXiv preprint arXiv:2003.13902. 2020.
14. Jiang M, Li Z, Zhang S, *et al.* Drug-target affinity prediction using graph neural network and contact maps. *RSC Adv* 2020;**10**(35):20701–12.
15. Wang X, Liu Y, Fan L, *et al.* Dipeptide frequency of word frequency and graph convolutional networks for DTA prediction. *Front Bioeng Biotechnol* 2020;**8**:267.
16. Stokes JM, Yang K, Swanson K, *et al.* Zohar Bloom-Ackermann, et al. A deep learning approach to antibiotic discovery. *Cell* 2020;**180**(4):688–702.
17. Liu X, Luo Y, Li P, *et al.* Deep geometric representations for modeling effects of mutations on protein-protein binding affinity. *PLoS Comput Biol* 2021;**17**(8):e1009284.
18. Gaudelet T, Day B, Jamasb AR, *et al.* Utilising graph machine learning within drug discovery and development. *Brief Bioinform* 05 2021;bbab159.
19. Wei GW. Persistent homology analysis of biomolecular data. *J Comput Phys* 2017;**305**:276–99.
20. Wei GW. Mathematics at the eve of a historic transition in biology. *Computational and Mathematical Biophysics* 2017;**5**(1).
21. Cang ZX, Wei GW. TopologyNet: Topology based deep convolutional and multi-task neural networks for biomolecular property predictions. *PLoS Comput Biol* 2017;**13**(7):e1005690.
22. Cang ZX, Wei GW. Integration of element specific persistent homology and machine learning for protein-ligand binding affinity prediction. In: *International journal for numerical methods in biomedical engineering*, page 10.1002/cnm.2914, 2017.
23. Nguyen DD, Xiao T, Wang ML, *et al.* Rigidity strengthening: A mechanism for protein–ligand binding. *J Chem Inf Model* 2017;**57**(7):1715–21.
24. Cang ZX, Wei GW. Integration of element specific persistent homology and machine learning for protein-ligand binding affinity prediction. *International journal for numerical methods in biomedical engineering* 2018;**34**(2):e2914.
25. Cang ZX, Mu L, Wei GW. Representability of algebraic topology for biomolecules in machine learning based scoring and virtual screening. *PLoS Comput Biol* 2018;**14**(1):e1005929.
26. Nguyen DD, Cang ZX, Wu KD, *et al.* Mathematical deep learning for pose and binding affinity prediction and ranking in D3R Grand Challenges. *J Comput Aided Mol Des* 2019;**33**(1):71–82.
27. Nguyen DD, Wei GW. AGL-Score: Algebraic graph learning score for protein-ligand binding scoring, ranking, docking, and screening. *J Chem Inf Model* 2019;**59**(7):3291–304.
28. Nguyen DD, Gao KF, Wang ML, *et al.* MathDL: Mathematical deep learning for D3R Grand Challenge 4. *Journal of computer-aided molecular design, pages* 2019;**1–17**.
29. Nguyen DD, Cang ZX, Wei GW. A review of mathematical representations of biomolecular data. *Phys Chem Chem Phys* 2020.
30. Puzyn T, Leszczynski J, Cronin MT. *Recent advances in QSAR studies: methods and applications*, Vol. **8**. Springer Science & Business Media, 2010.
31. Lo YC, Rensi SE, Torng W, *et al.* Machine learning in chemoinformatics and drug discovery. *Drug Discov Today* 2018;**23**(8):1538–46.
32. Bajorath J. *Chemoinformatics: concepts, methods, and tools for drug discovery*, Vol. **275**. Springer Science & Business Media, 2004.
33. Nguyen DD, Gao K, Chen J, *et al.* Unveiling the molecular mechanism of SARS-CoV-2 main protease inhibition from 137 crystal structures using algebraic topology and deep learning. *Chem Sci* 2020;**11**(44):12036–46.
34. Nguyen DD, Wei GW. DG-GL: Differential geometry-based geometric learning of molecular datasets. *International journal for numerical methods in biomedical engineering* 2019;**35**(3):e3179.
35. Liu ZH, Li Y, Han L, *et al.* PDB-wide collection of binding data: current status of the PDBbind database. *Bioinformatics* 2015;**31**(3):405–12.
36. Wee JJ, Xia K. Forman persistent ricci curvature (FPRC) based machine learning models for protein-ligand binding affinity prediction. *Briefings in Bioinformatics, in press* 2021.
37. Qi CR, Hao S, Mo K, *et al.* Pointnet: Deep learning on point sets for 3d classification and segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, 652–60.
38. Qi CR, Yi L, Su H, *et al.* Pointnet++: Deep hierarchical feature learning on point sets in a metric spacearXiv preprint arXiv:1706.02413. 2017.
39. Sagi O, Rokach L. Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 2018;**8**(4):e1249.