Molecular Diversity
https://doi.org/10.1007/s11030-025-11451-9

RESEARCH

LightDTA: lightweight drug-target affinity prediction via random-walk network embedding and knowledge distillation

Xiaoyu Huang[1] · Xiangpeng Bi[1] · Nianwen Xing[1] · Wenjian Ma[1] · Huasen Jiang[1] · Qing Cai[2] · Weigang Lu[1] · Fei Yang[2] · Zhiqiang Wei[1,3] · Shugang Zhang[1]

Received: 24 September 2025 / Accepted: 22 December 2025

Abstract

Accurately predicting drug targeting affinity is crucial in the field of drug discovery. With the rapid development of artificial intelligence, many deep learning methods have been proposed for drug target affinity prediction tasks. However, most existing methods rely heavily on a detailed description of biochemical attributes of inputs; besides, the model architecture is getting increasingly complex just to achieve a slight performance gain. Together, these poses great challenges for real-world employments and applications. This study proposes a new lightweight framework, LightDTA, which combines knowledge distillation and random walk algorithms to predict drug target affinity. It adopts a lightweight network-based protein representation and eliminates the tedious process of collecting detailed biochemical properties. A knowledge distillation framework is further introduced to enrich molecular-level knowledge and enhance predictive capability while not affecting the model efficiency. Comprehensive experiments show that LightDTA achieves state-of-the-art performance in both classification and regression tasks, with only 61% of the memory requirements of the suboptimal baseline model. It also achieves a 7 × speedup in inference time. Therefore, the proposed method offers a highly efficient and accurate model for real-world prediction of drug-target affinities. The code for LightDTA is publicly available at: https://github.com/Huang-zilin/LightDTA-final.

Keywords Drug-target affinity prediction · Graph neural network · Drug discovery · Deep learning

Xiaoyu Huang and Xiangpeng Bi have contributed equally to this work.

✉ Shugang Zhang
zsg@ouc.edu.cn

Xiaoyu Huang
huangxiaoyu@stu.ouc.edu.cn

Xiangpeng Bi
bixiangpeng@stu.ouc.edu.cn

Nianwen Xing
xingnianwen@stu.ouc.edu.cn

Wenjian Ma
mwj4251@stu.ouc.edu.cn

Huasen Jiang
Jianghuasen@stu.ouc.edu.cn

Qing Cai
cq@ouc.edu.cn

Weigang Lu
luweigang@ouc.edu.cn

Fei Yang
feiyang@sdu.edu.cn

Zhiqiang Wei
weizhiqiang@ouc.edu.cn

[1] College of Computer Science and Technology, Ocean University of China, Qingdao 266404, Shandong, China

[2] Electrical and Information Engineering, Shandong University at Weihai School of Mechanical, Shandong University, Weihai 264209, Shandong, China

[3] College of Computer Science and Technology, Qingdao University, Qingdao 266071, Shandong, China

Published online: 07 January 2026

Introduction

Conventional drug development process is usually accompanied by time-consuming, laborious, and expensive biochemical experiments [1]. Estimates suggest that discovering a new drug takes more than 10 years and costs up to nearly $2.7 billion on average [2], and a substantial proportion of candidates fail in clinical trials. Essentially, drug discovery is to identify drugs, normally small molecules, that are able to interact with particular target proteins [3]. However, considering the enormous number of drug candidates, it is unpractical to screen them via experimental measurements. Therefore, it is essential to develop efficient and accurate in silico drug screening methods. In this regard, early attempts have relied on molecular dynamics simulation and molecular docking [4]; however, they are still far from satisfactory due to the expensive computational costs and the poor extensibility to novel proteins without known structures.

With the rapid development of artificial intelligence, deep learning-based approaches for drug screening, specifically in predicting drug-target affinity (DTA), have gained significant momentum in recent years. In terms of methodology, these methods can be roughly divided into two categories, namely the sequence-based approaches and the structure-based approaches. Since the sequence data, including residue sequence of proteins and simplified molecular input line entry specification (SMILES) of drugs, is easy to obtain, it has driven the early deep learning-based DTA prediction methods. For examples, DeepDTA [5] extracts latent features from drug SMILES and protein sequences via convolutional neural networks (CNN), without incorporating additional structural features. As an extension, WideDTA [6] utilizes four parallel CNNs to encode protein sequence, ligand SMILES, protein domains and motifs, and maximum common substructure words to predict binding affinity. In addition to CNN, other sequential methods like long short-term memory (LSTM) [7] and recurrent neural networks (RNN) [8] have also been used as encoders in DTA prediction. Obviously, these sequence-based methods extract features solely from the sequential modality of molecules, which neglect the structural characteristics and therefore lead to suboptimal representation.

The drawback of sequence-based approaches has led to the development of structure-based methods. Two-dimensional (2D) graphs are the most common way of describing the input drugs and proteins, where the drug naturally corresponds to a molecular graph [9] and the protein is converted to a contact map by setting a predefined threshold of distance between residues [10]. Accordingly, graph neural networks (GNN) and its variants can be employed to learn structural representations for a better predictive performance. For example, SAG-DTA [11] utilizes self-attention pooling operation on drug molecular graphs to differentiate the contributions of atoms. MGraphDTA [12] uses multiple multiscale blocks, with each block containing five GCN layers along with dense connections among them. MvGraphDTA [13], which introduces not only the node graphs but also the inverted line graphs of drugs and targets for prediction. Building upon the 2D graphs, recent works further extend the representation to three-dimensional (3D) space for a more fine-grained geometric learning [14], or to macroscopic semantics like drug-protein bipartite networks [15] and protein–protein networks [16]. For example, KDB-Net [17] utilizes a 3D-equivariant GCN to learn geometric representations of the drug molecule, while the protein geometric information is incorporated via a set of 3D-invariant geometric features.

In the broader field of bioinformatics, predictive methods that extract and fuse multi-level data have achieved notable success in improving the performance of complex classification tasks [18–20]. For instance, the model proposed by Han et al. employs hybrid feature extraction techniques for Cyber Threat Intelligence capability assessment, demonstrating good performance in classification tasks and providing a referential framework for hybrid feature engineering [21]. The XGBoost Liver model [22] integrates ranking and statistical projection-based strategies for liver disease prediction, enabling not only feature integration but also global interpretability analysis through Fisher scoring, which selects optimal features for training by evaluating their contributions to the model. The Deep ProBind model proposed by Khan et al. [23] leverages Transformer architecture to fuse sequence and structural information for high accuracy binding protein prediction, and applies the SHAP interpretability algorithm to select optimal hybrid features, offering a solution that balances performance and interpretability for protein interaction classification. The 5 meth Uri predictor introduced by Almusallam et al. [24] combines dinucleotide and trinucleotide auto/cross covariance features with six physicochemical parameters to construct feature vectors, whose high accuracy and reliability have been thoroughly verified in independent tests.

Despite the good performance that current methods have achieved, there are two challenges to be addressed. (1) *Burden of collecting intact molecular attributes*. To achieve a comprehensive description of the input proteins and molecules, current methods try to gather myriad molecular properties, such as evolutionary features [25], structural features like probabilistic residue-residue contact energies, statistical potential, and protein binding domains [26, 27], and other physicochemical properties including composition, polarity and molecular volume [28, 29]. These pose great challenge upon actual model deployment since these attributes

are neither easy to collect nor reliable due to the absence of a well-recognized data source. (2) *Difficulty in balancing efficiency and accuracy*. Early unimodal and single-scale approaches exhibit clear shortcomings in prediction accuracy. Accordingly, multimodal and multiscale representation-based methods have been continuously raised in recent years. In this aspect, advanced techniques like cross-attention mechanism and contrastive learning are frequently adopted for cross-modality fusion [30–32]. For the multi-scale approaches, hierarchical graphs are commonly used, where the global network is responsible for modeling macroscopic biological interactions (e.g., PPI) while each of its node is also a graph denoting the molecular structure of a protein [16, 33]. Although these methods have achieved impressive performance, their substantial computational complexity and storage requirements pose significant challenges for practical deployment, particularly in meeting real-time demands on resource-constrained devices. Aiming at the above problems, we proposes a knowledge distillation-driven **light**weight method for the **DTA** prediction, termed **LightDTA**. The main contributions of this work are summarized as follows:

(1) A lightweight network-based protein representation learning method is proposed, which relies solely on the topological structure of protein interaction networks, eliminating the tedious process of collecting detailed biochemical properties to construct protein descriptors.
(2) A knowledge distillation framework is introduced to transfer molecular-level knowledge from a comprehensive but computational expensive teacher model, which further enhance the predictive capability of the model while not affecting its efficiency.
(3) Comprehensive experiments show that, despite using a lightweight architecture, LightDTA achieves state-of-the-art (SOTA) performance in both classification and regression tasks. Furthermore, LightDTA demonstrates exceptional generalization capability and practical performance across more challenging scenarios, including both temporal and structural split-based cold-start settings, as well as an external COVID-DTA dataset. Therefore, it offers a highly efficient and accurate model for real-world employment.

Methods

To describe the computational task and the proposed method more precisely, we first formulate the DTA prediction task by giving mathematical descriptions of the model inputs and expected outputs. We then present the detailed construction process of graph structures of proteins and drugs. In the last section, we introduce in detail the three components in the framework of LightDTA, namely the drug representation learning module, the random walk-based protein representation learning module, and the knowledge distillation module.

Problem Statement

The DTA prediction is a type of regression task, where the ground truth y is a continuous value indicating the binding strength (or affinity) between a pair of drug molecule and target protein<drug, protein>. While for the CPI (compound-protein interaction) prediction as a binary classification task, the ground truth y is discretized into a binary label being either "1" indicating a positive interaction for the input<drug, protein>, or "0" indicating no interaction between them.

In this work, the input<drug, protein>is processed into two graphs, including a molecular graph $\mathcal{G}_k = \{V_k, E_k\}$, and a protein–protein interaction network $\mathcal{G}_{ppi} = \{V_{ppi}, E_{ppi}\}$. Note that, instead of treating a protein as a graph by constructing its contact map, we treat the protein itself as a node in $\mathcal{G}_{ppi}$. The task is to learn a biochemical attribute-free protein representation r_p and combine it to the drug representation r_d (learned from $\mathcal{G}_k$) for the final prediction $\hat{y}_{\langle d,p\rangle}$.

Descriptions of drugs and proteins

Protein–protein interaction network (PPI)

To construct the PPI network, we first identified all target proteins involved in this study and retrieved their interactions from STRING [34]. Specifically, two protein nodes with an interaction score higher than a preset value are considered to have interactions, and an edge is put between them. Accordingly, the PPI network $\mathcal{G}_{ppi} = \{\mathcal{V}_{ppi}, \mathcal{E}_{ppi}\}$ can be built for the given protein sets, with the adjacency matrix being $\mathbf{A}_{ppi} \in \{0,1\}^{\mathrm{N}_{ppi} \times \mathrm{N}_{ppi}}$ and the node feature matrix being $\mathbf{X}_{ppi} \in \mathbb{R}^{\mathrm{N}_{ppi} \times \mathrm{D}_{ppi}}$. N_{ppi} is the number of protein nodes in $\mathcal{G}_{ppi}$, and $\mathrm{D}_{ppi} = 64$ denotes the dimension of the initial protein descriptor, which in our case is obtained via random walk.

Drug molecular graph

The drug molecule is naturally a graph if treating atoms as nodes and chemical bonds as edges. The open-source tool RDKit [35] is used to convert the drug SMILES to the corresponding molecular graph $\mathcal{G}_d = \{\mathcal{V}_d, E_d\}$. The node set $\mathcal{V}_d$ contains all the atoms v_i of the molecule (i.e.,=$|\mathcal{V}_d|$ N_{atom}), while the edge set $\mathcal{E}_d$ contains the covalent bonds e_{ij} (if

Table 1 Atom node descriptors in drug graph

Descriptor	Dimensions
Atom symbol	44
Degree of the atom	11
Total count of hydrogen atoms bonded to the heavy atom	11
Number of hidden hydrogen atoms bonded to the heavy atom	11
Whether the atom is aromatic	1
Total	78

Table 2 Descriptor features of residue-level in protein graph

Descriptor	Dimensions
Residue symbol (one-hot)	21
Whether the residue is aliphatic	1
Whether the residue is aromatic	1
Whether the residue is polar neutral	1
Whether the residue is acidic (negatively charged)	1
Whether the residue is basic (positively charged)	1
Residue weight	1
Dissociation constant for the –COOH group (-log)	1
Dissociation constant for the $-NH_3$ group (-log)	1
Dissociation constant for any other group in the molecule (-log)	1
pH at the isoelectric point	1
Hydrophobicity of the residue (pH = 2)	1
Hydrophobicity of the residue (pH = 7)	1
All	33

exists) between the i-th and j-th atoms. Therefore, the adjacency matrix of $\mathcal{G}_d$ is represented by the binary undirected graph $\mathbf{A}_d \in \{0,1\}^{|\mathcal{V}_d|\times|\mathcal{V}_d|}$, and the feature matrix of nodes is represented by $\mathbf{X}_d \in \mathbb{R}^{|\mathcal{V}_d|\times 78}$. For each node atom, the 78-dimensional vector describes certain atomic properties shown in Table 1.

Target protein graph

As for the proteins, each of them was converted into a 2D topological graph by regarding amino acid residues as nodes to obtain residue-level features, while the edges were generated by calculating the residue contact map. Following the previous work [10], we set the contact distance threshold as 8 Å to decide whether two residues contact with each other. For those proteins without experimentally solved structures, AlphaFold2-predicted structures were used as substitutes since previous studies have demonstrated a comparable performance of AlphaFold-predicted to real structures [36]. Finally, each residue node in the protein graph was assigned a 33-dimensional descriptor representing basic residue attributes, as detailed in Table 2.

The framework of LightDTA

The framework of LightDTA is shown in Fig. 1. Generally, it retains the standard dual-branch architecture, i.e., the input drug and protein are encoded separately before being concatenated for final prediction. In this study, the particular contributions are focused on the protein representation branch, where a protein–protein interaction network is constructed and a lightweight random walk module is introduced to learn the protein embedding. Besides, a knowledge distillation strategy is adopted to further enrich the protein representation. A hierarchical model with abundant protein molecular features is leveraged as the teacher model, and thus the fine-grained protein knowledge can be distilled into LightDTA without affecting its efficiency.

Drug representation learning

Graph convolutional networks (GCN) are utilized to extract drug features from the input molecular graph $\mathcal{G}_d$. For each atom node in the graph, a typical message-passing operation is conducted to aggregate features from the surrounding neighbors:

$$\mathbf{h}_i^{(l+1)} = \sigma\left(\mathbf{W}_1\mathbf{h}_{d,i}^{(l)} + \mathbf{W}_2 \sum_{j\in\mathcal{N}(i)} \mathbf{h}_{d,j}^{(l)}\right) \tag{1}$$

where $\mathbf{W}_1, \mathbf{W}_2$ are two learnable weights, $\mathcal{N}(i)$ is the set of neighboring nodes of vertex i. $\sigma(\cdot)$ indicates the ReLU activation function. $\mathbf{h}_{d,i}^{(l)}$ indicates the embedding of the i-th node of the molecular graph at the l-th GCN layer, and the initial $\mathbf{h}_{d,i}^{(0)} = \mathbf{x}_{d,i}$. Three layers of graph convolution are employed, and the updated atom node features are aggregated and fed to an MLP block to generate the final graph-level drug representation $\mathbf{r}_d$ via the following readout layer:

$$\mathbf{r}_d = MLP\left(\frac{1}{N_{atom}} \sum_{i=1}^{N_{atom}} \mathbf{h}_{d,i}^{(3)}\right) \tag{2}$$

The readout phase aggregates the vertex embeddings into a unified graph embedding that transforms local features in the molecular graph into global features.

Protein representation learning via random walk on the PPI network

Random walk is a highly efficient technique that often being utilized to generate graph embeddings from the graph topology. Inspired by the natural language processing, random walk methods sample sequences of nodes from a graph (or network) so that to turn the graph into multiple ordered

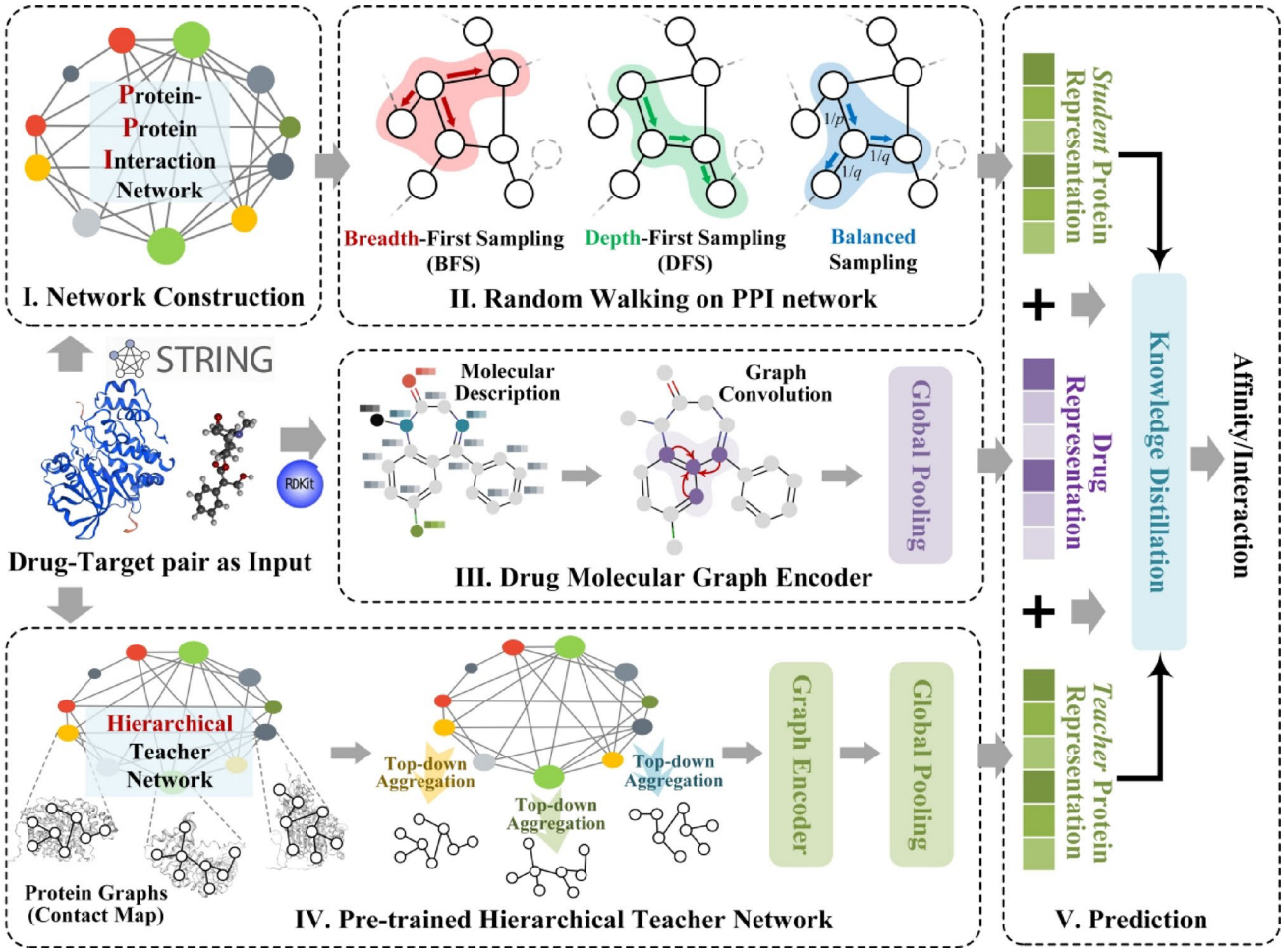

Fig. 1 The overall framework of LightDTA

sequences of nodes for representation learning. The proposed LightDTA leverages random walk to learn the initial protein node embeddings by applying it to the constructed PPI network $\mathcal{G}_{ppi}$. Overall, the optimization objective can be denoted as:

$$\max_{f} \sum_{u \in \mathcal{V}} \frac{\sum_{v \in \mathcal{N}(u)} f(u) \cdot f(v)}{\log\left(\sum_{w \in \mathcal{V}} \exp(f(u) \cdot f(w))\right)} \tag{3}$$

where $f(\cdot)$ is the embedding function to be learned. u, v, and w are protein nodes in the PPI network, and $\mathcal{N}(u)$ indicates the sampled neighboring nodes of the node u. The function is optimized via the skip-gram architecture.

In particular, the neighborhood set for a node is obtained via random walk and depends on specific sampling strategies. In this work, we introduce the neighborhood sampling strategy in node2vec [37] and devise three schemes. Formally, the ordered sentences of nodes are sampled from the following distribution:

$$P(s_i = x | s_{i-1} = v, s_{i-2} = u) = \begin{cases} \frac{\alpha_{pq}(u,x)}{Z} & \text{if } e_{vx} \in E_{ppi} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

The above equations define a 2nd order Markov process, in which the choice of which node to sample depends on the two preceding steps. Z is the normalizing constant, u and v are the sampled nodes at the previous two steps. $\alpha_{pq}(u, x)$ is the search bias indicating how the node at the current step will be sampled. It is parameterized by p and q, and calculated as follows:

$$\alpha_{pq}(u, x) = \begin{cases} \frac{1}{p} & \text{if } d_{ux} = 0 \\ 1 & \text{if } d_{ux} = 1 \\ \frac{1}{q} & \text{if } d_{ux} = 2 \end{cases} \tag{5}$$

The parameter p is the *return parameter* controlling the likelihood of revisiting a previously sampled node, while q is the *in–out parameter* allowing the search to go further or around the center. The parameter q affects the tendency to node traversal towards internal and external nodes when performing node traversal. When $q > 1$, the random walk

is biased towards nodes close to node u. This kind of walk provides a sufficiently localized view around the start node and approximates the breath-first sampling (BFS) strategy in the sense that the sample consists of nodes in the local domain. In contrast, when $q < 1$, walking is more likely to visit nodes further away from node u, reflecting a depth first sampling (DFS)-like strategy of outward exploration. As for the return parameter p, it controls the possibility of revisiting already visited nodes immediately when performing a node traversal. Setting the parameter p to a higher value ($> \max(q, 1)$) means that already visited nodes are unlikely to be sampled in the next two steps unless the next node in the walk has no other neighbors. On the contrary, a lower parameter p ($< \min(q, 1)$) causes the walk to take a step backward, making the path of the walk more "local" and closer to the starting node u. This setting encourages moderate exploration of the network while avoiding two-step redundancy in the sampling process.

Depending on the values of p and q, there will be numerous sampling strategies. Here we focused on three representative ones among them, namely *breath-first sampling* (BFS), *depth-first sampling* (DFS), and *balanced sampling*.

- *Breath-first sampling* (BFS). In this setting, we set $p = 0.5$ and $q = 3$ so that the random walk is biased towards nodes around the starting node.
- *Depth-first sampling* (DFS). In this setting, we set $p = 3$ and $q = 0.5$ so that the random walk tends to visit remote nodes from the starting point.
- *balanced sampling*. In this setting, we set $p = 0.25$ and $q = 0.5$ to achieve a balanced sampling upon random walking. The values are determined via hyperparameter optimization.

Through the above random walk operation, the initial protein node embedding $\mathbf{x}_{p,i}$.can be obtained. After that, a three-layer GCN encoder is applied to the PPI netowork $\mathcal{G}_{ppi}$, with each layer denoted as:

$$\mathbf{h}_i^{(l+1)} = \sigma\left(\mathbf{W}_3\mathbf{h}_{p,i}^{(l)} + \mathbf{W}_4 \sum_{j\in\mathcal{N}(i)} \mathbf{h}_{p,j}^{(l)}\right) \tag{6}$$

where $\mathbf{W}_3, \mathbf{W}_4$ are learnable weights, $\mathcal{N}(i)$ is the set of interacting proteins of protein i. $\mathbf{h}_{p,i}^{(l)}$ indicates the embedding of the i-th node at the l-th GCN layer, and the initial $\mathbf{h}_{p,i}^{(0)} = \mathbf{x}_{p,i}$. An MLP block is applied to each node feature (rather than the pooled feature) to get the final representation:

$$\mathbf{r}_{p,i} = MLP\left(\mathbf{h}_{p,i}^{(3)}\right) \tag{7}$$

Till now both drug and protein representations have been obtained. They are concatenated for the final prediction:

$$\hat{y}_{\langle d,p\rangle} = MLP(\mathbf{r}_d||\mathbf{r}_p) \tag{8}$$

Knowledge distillation

The above random walk on the PPI network provides the protein profiles in terms of the topology of interaction network; however, it lacks fine-grained protein knowledge such as the intramolecular characteristics and chemical properties. To further enrich the protein representation while maintaining its lightweight property, a knowledge distillation scheme is introduced. Knowledge distillation is a model compression technique that trains a small-scale *student* model by transferring knowledge from a larger *teacher* model [38], so that to achieve competitive or even better performance. Considering that there is still a lack of fine-grained protein features, we introduce a hierarchical network as the teacher model where each node is further characterized by protein graphs (i.e., contact maps), as shown in Fig. 2.

Specifically, the teacher model adopts a hierarchical fusion strategy for protein representation learning, which integrates both high-level features (i.e., protein–protein interactions) and low-level features (i.e., inherent molecular characteristics). The model first uses a GCN encoder block to learn from the PPI network $\mathcal{G}_{t,ppi}$, obtaining node embeddings h_i that contain high-order topological semantic information as shown in the following equation:

$$\mathbf{h}_i^{(l+1)} = \sigma\left(\mathbf{W}_5\boldsymbol{h}_{t,i}^{(l)} + \mathbf{W}_6 \sum_{j\in\mathcal{N}(i)} \mathbf{h}_{t,j}^{(l)}\right) \tag{9}$$

This encoder block includes two GCN layers and a DropEdge regularization term. Here, $\mathbf{W}_5$ and $\mathbf{W}_6$ are lernable weights, $\mathcal{N}(i)$ is the set of interacting proteins of protein i, $\mathbf{h}_{t,i}^{(l)}$ indicates the embedding of the i-th node at the l -th GCN layer. Then the obtained PPI graph node embedding features $\mathbf{h}_{t,i}^{(2)}$ are propagated to all residues within the corresponding low-order protein graph, achieving feature fusion as follows:

$$\mathbf{x}'_{t,p} = \left[\mathbf{x}_{t,p} \oplus \mathbf{h}_{t,i}^{(2)}\right] || \left[\mathbf{x}_{t,p} \ominus \mathbf{h}_{t,i}^{(2)}\right] \tag{10}$$

where $\oplus$ and $\ominus$ denote elementwise addition and elementwise subtraction on vectors, $||$ denotes vector concatenation, and $\mathbf{x}_{t,p} \in \mathbb{R}^{1\times \mathrm{D}_p}$ denotes the initial residue feature. After performing this feature combination operation for all residues in the protein graph, we obtain a new feature matrix

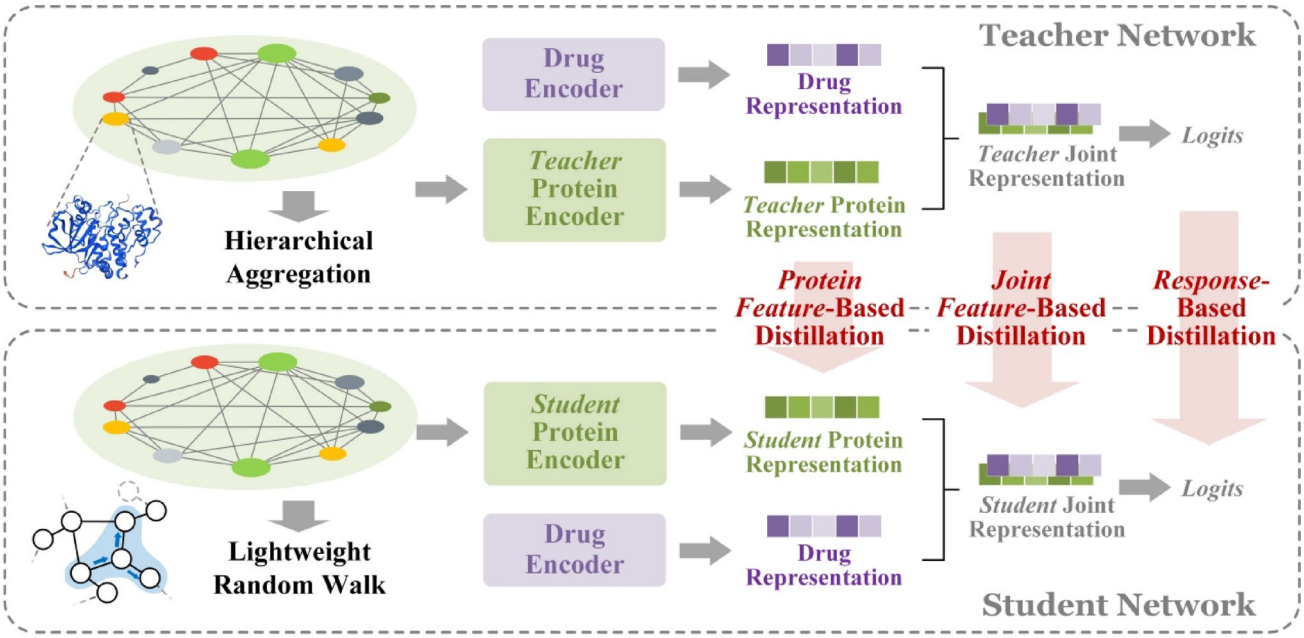

Fig. 2 Schematic of different knowledge distillation strategies

$\mathbf{x}_{t,p}\prime$, to which a GCN encoder is then applied to derive the embedding q.

$$\mathbf{h}_i^{\prime(l+1)} = \sigma\left(\mathbf{W}_7\mathbf{h}_{t,i}^{\prime(l)} + \mathbf{W}_8 \sum_{j\in\mathcal{N}(i)} \mathbf{h}_{t,j}^{\prime(l)}\right) \tag{11}$$

$\mathbf{W}_7$ and $\mathbf{W}_8$ are lernable weights, $\mathcal{N}(i)$ is the residua set of protein, $\mathbf{h}'^{(l)}_{t,i}$ indicates the embedding of the i-th node at the l-th GCN layer and $\mathbf{h}'^{(0)}_{t,j} = \mathbf{x}_{t,p}\prime$ is the initial protein node embedding. Then, we performed average pooling on them to obtain the protein representation q from $\mathbf{h}'^{(3)}_{t,j}$.

$$q = \frac{1}{\mathrm{N}_p}\sum_{i=1}^{\mathrm{N}_p} \mathbf{h}_{t,j}^{\prime(3)} \tag{12}$$

Finally, the graph embedding $\boldsymbol{q}$ can be transformed into the final protein representation $\boldsymbol{p}$ via a linear transformation layer, as shown in the following equation, where $\mathbf{W}$ is the linear parameter and b is a bias term.

$$p = Wq + b \tag{13}$$

The drug branch and the feature concatenation are the same as that in the student model and are not elaborated here. Depending on the distillation scheme, the learned features in different stages of the teacher model are fed into the student model.

Depending on the level of distillation, we devise three candidate strategies in LightDTA, namely the response-based strategy, the joint feature-based strategy, and the protein feature-based strategy.

(1) Response-based distillation strategy

Response based distillation learning typically refers to training student models through the neural response of the last layer of the teacher model. By directly imitating the final prediction results of the teacher model and optimizing the loss function to match the output of the student model with the teacher model, the goal of model compression and performance improvement can be achieved. The calculation of the loss function is based on the output of the last fully connected layer of a large deep learning model, which utilizes the differences in the Logits layer to calculate the distillation loss. For the DTA task, the loss of the response-based knowledge distillation can be expressed as:

$$L_{dis} = L\left(\hat{y}_t, \hat{y}_s\right) \tag{14}$$

where $\mathcal{L}(\cdot)$ represents the differential loss between the logits of teacher model and student model, $\hat{y}_t$ and $\hat{y}_s$. For the CPI task, the model outputs are binary labels; therefore, the distillation loss is calculated using soft objectives estimated by a softmax function.

$$p\left(z_i, T\right) = \frac{\exp\left(\frac{z_i}{T}\right)}{\sum_j \exp\left(\frac{z_j}{T}\right)} \tag{15}$$

where z_i is the logit for the i-th binary class, and a temperature factor T is introduced to control the importance

of each soft target. Comparing to the hard label, the soft target contains the informative knowledge from the teacher model. Accordingly, the distillation loss for soft-targets can be rewritten as:

$$L_{dis} = L\left(p\left(z_t, T\right), p\left(z_s, T\right)\right) \tag{16}$$

The distillation loss is then integrated with the loss supervised by the ground truth label, and a weighting parameter α is introduced to control the relative contribution of distillation:

$$L_{\text{total}} = \alpha L_{dis} + (1-\alpha) L_{true} \tag{17}$$

(2) Joint feature-based distillation strategy

In addition to the response-based distillation, the student model can also imitate the latent features generated in the intermediate layer of the teacher model. Specifically, the concatenated feature maps before MLP processing in the teacher model, which contains both drug and protein information, are utilized for distillation. For this strategy, the distillation loss function for DTA and CPI tasks can be unified and calculated as:

$$L_{dis} = L\left(\phi_t\left(f_t\left(x\right)\right), \phi_s\left(f_s\left(x\right)\right)\right) \tag{18}$$

Among them, $f_t\left(x\right)$ and $f_s\left(x\right)$ are the feature maps of the hidden layer of the teacher and student models, respectively. $\Phi_t\left(\cdot\right)$ and $\Phi_s\left(\cdot\right)$ are the transformation functions for aligning the feature space.

(3) Protein feature-based distillation strategy

Since the student model differs from the teacher model primarily in the protein branch, we conceive the third distillation strategy where only the protein feature is extracted for knowledge transfer. Specifically, this strategy first extracts protein features aggregated from the teacher model's hierarchical network of proteins, and these features are then used as supervision to guide the student model (in particular, the random walk module) in learning protein representation.

Table 3 Summary of the benchmark datasets

Dataset	#Drugs	#Proteins	#Binding Entries	#PPI*	Task Type
Davis	68	442	30,056	7049	Regression
KIBA	2111	229	118,254	5091	Regression
Human	2726	2001	3369(+)/3359(−)	92,700	Classification

*PPI indicates the number of interaction edges in the protein–protein interaction graph

Similar to the joint feature-based distillation, a loss function can be devised between the two protein features obtained from student and teacher models. Compared to the aforementioned two strategies, the current one focus directly on the different component between the two models.

Results

Benchmark datasets

Three widely used datasets, namely Davis [39], KIBA [40], and Human [41] datasets as summarized in Table 3, are used to benchmark the baseline models and LightDTA. Davis and KIBA datasets are two widely acknowledged datasets for DTA prediction tasks in recent years due to its standardized and high-quality data samples. As regression tasks, these two datasets provide a wealth of drug target binding affinity values, measured by dissociation constant (K_{d}) and KIBA score, enabling an effective model training and evaluation. Meanwhile, the Human dataset serves as a CPI binary classification task that differs from the above two datasets. It contains experimentally determined binary labels indicating the interaction relationship for each drug-target pair. Evaluation on different types of tasks help to realize the generalizability and robustness of models.

Evaluation metrics

For the datasets of regression task, i.e., Davis and KIBA, three commonly used metrics including MSE, CI, and r_m^2 are adopted to evaluate the model performance. MSE measures the predictive performance of a model in terms of the square error between the predicted value and the actual value:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{19}$$

where $\hat{y}_i$ is the predicted value of the i-th sample, and y_i is the ground truth. n indicates the number of samples in datasets. Next, the metric CI (consistency index) is calculated to measure the degree of consistency between pairwise comparisons:

$$CI = \frac{1}{Z}\sum_{y_i > y_j} h\left(\hat{y}_i - \hat{y}_j\right) \tag{20}$$

where Z is a normalized constant and $\mathrm{h}\left(x\right)$ is a step function defined as:

$$h(x) = \begin{cases} 0, & x < 0 \\ 0.5, & x = 0 \\ 1, & x > 0 \end{cases} \tag{21}$$

Besides, the metric r_m^2 is employed to access the generalization ability, which is calculated as:

$$r_m^2 = r^2 (1 - r_0) \tag{22}$$

where r is the square of the correlation coefficient with intercept, and r_0 is the correlation coefficient without intercept.

For the Human dataset as a classification task, the following metrics are used:

$$Precision = \frac{TP}{TP+FP} \tag{23}$$

$$Recall = \frac{TP}{TP+FN} \tag{24}$$

where TP, FP, and FN represent the number of true positive, false positive, and false negative samples, respectively. In addition, the receiver operating characteristic curve area (AUROC) is also used for further comparisons.

Experimental setup

Training detail and data splitting

In addition to the three benchmark datasets, we also evaluated the model performance using two external datasets: Metz and COVID. Three splitting schemes were employed in this study: random splitting, cold-start splitting, and scaffold-based splitting. Under the random-split setting, the training-to-test ratio was set to 5:1 for the Davis and KIBA datasets, while a 4:1 ratio was used for the Metz and COVID datasets. For instance, the Metz dataset was partitioned into a training set containing 80% (28,207 samples) of the total data and a test set comprising the remaining 20% (7,052 samples). Note that for the Davis dataset in particular, its training set was used to determine the optimal hyperparameters. Therefore, it was further divided into five folds, with each fold acting as validation set in turn, aka cross validation.

Meanwhile, the binary classification task, i.e. the Human dataset, also followed a 4:1 random splitting for training and test sets. However, considering the relatively small size of this dataset (3,369 positive and 3,359 negative samples), we performed cross test evaluation by repeating the training process five times and used each fold in turn as the test set. The average results from the five folds were taken as the final model performance for this dataset.

The aforementioned random splitting strategy is the most popular one due to its ease of implementation. However, the random splitting may lead to overly optimistic results due to the shared drugs and targets in training and test sets. In other words, information leakage will occur if a drug (or target) in the training set appears again in the test set, which makes it easier for the model to achieve high scores by memorizing the particular drug rather than understanding the hidden binding pattern. Accordingly, to further validate the effectiveness of the proposed model, we also evaluated two additional splitting scenarios on the Davis dataset in addition to the random splitting, namely the cold-start scenario and the scaffold-based scenario.

The cold-start scenario is a condition where the drug molecules and proteins in the test set are never exposed during training. We regard this as one of the temporal splits, since models are always trained on currently available drugs and targets, while they never see the newly developed drugs or newly discovered targets in later years.

For the scaffold-based splitting scenario, we adopted the Murcko scaffold [42] partitioning protocol that in line with previous studies [43–45], which decomposes drug molecules into four fundamental structural components, including ring systems, linkers, scaffolds, and side chains, as shown in Fig. 3. In our splitting strategy, compounds were grouped based on their core scaffolds, ensuring that molecules sharing identical scaffolds were assigned to the same subset (either for training or testing). This approach effectively separates structurally distinct compounds, thereby simulating real-world scenarios in which models encounter novel chemical scaffolds.

Hyperparameter settings

The hyperparameters in LightDTA are listed in Table 4. To avoid excessive computational costs and overfitting risks, we only fine-tuned those hyperparameters that are most relevant to the core modules (random walk and distillation) in LightDTA. The fine-tuning was conducted via five-fold cross validation on the training set of the Davis dataset, and the determined optimal hyperparameters were used consistently across the other datasets unless otherwise stated. A more detailed analysis on the impact of these hyperparameters is presented in below.

Performance of different random walk methods

The in–out parameter q and return parameter p are the two most critical parameters responsible for guiding the random walk strategies. In section **Protein representation learning via random walk on the PPI network**, we have described three random walk variants in terms of sampling strategies (BFS, DFS, and balanced sampling) in LightDTA. In this section, we evaluate the performance of the three variants. For comparison, we also test the performance of other two

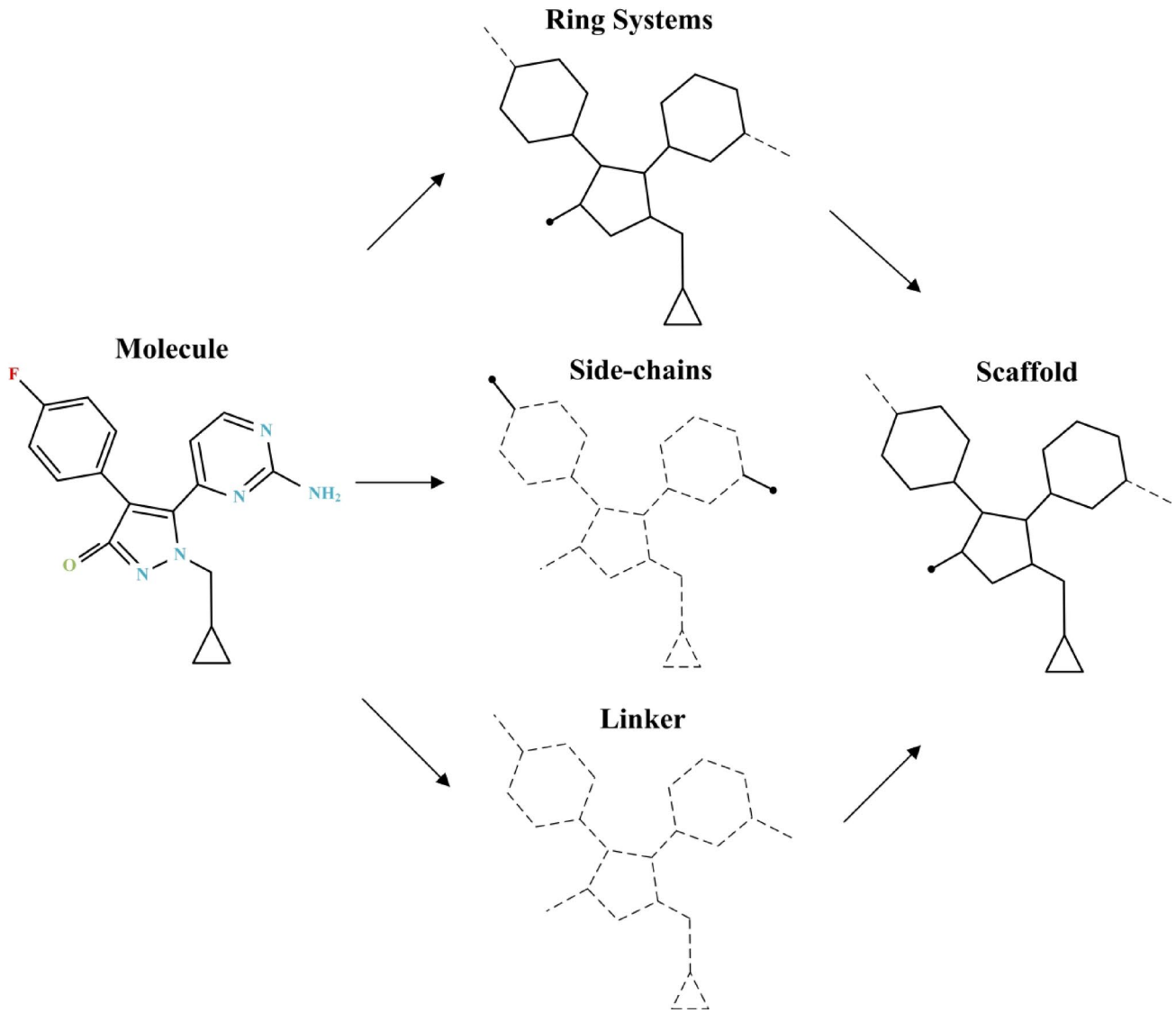

Fig. 3 Molecule scaffold decomposition methodology

Table 4 Hyperparameter settings of LightDTA

Hyperparameter	Value(s)
Optimizer	Adam
Learning rate	0.0005
Epoch	2000
Combined_score C	0.04
The number of FC layers in classifier	2
Batch size	512
In–out parameter q[a]	0.5
Return parameter p[a]	0.25
Random walk strategies[a]	DeepWalk/Node2vec/LINE/BFS/DFS
Distillation strategies[a]	Protein Feature-Based Distillation/ Joint Feature-Based Distillation / Response-Based Distillation

[a]Hyperparameters to be fine-tuned in this study

common random walk methods, namely DeepWalk [46] and Line [47]. The evaluation results are shown in Fig. 4.

It can be observed that the random walk with balanced sampling obtained the best score across all datasets, with MSE of 0.190 on Davis and 0.121 on KIBA, and AUROC of 0.988 on Human. DFS, as a biased variant that favors exploring more distant nodes, achieved poor performance due to the inclusion of noise when wandering too far. In contrast, when restricted to the local nodes around the starting one, the model does not work well either due to the insufficient understanding of the global structure or inability to recognize potential connections with other nodes. As for the other two methods, DeepWalk achieved comparable performance to our balanced sampling variant, i.e., MSE of 0.192 and 0.123 on Davis and KIBA, and AUROC of 0.988 on Human. LINE was slightly inferior to DeepWalk,

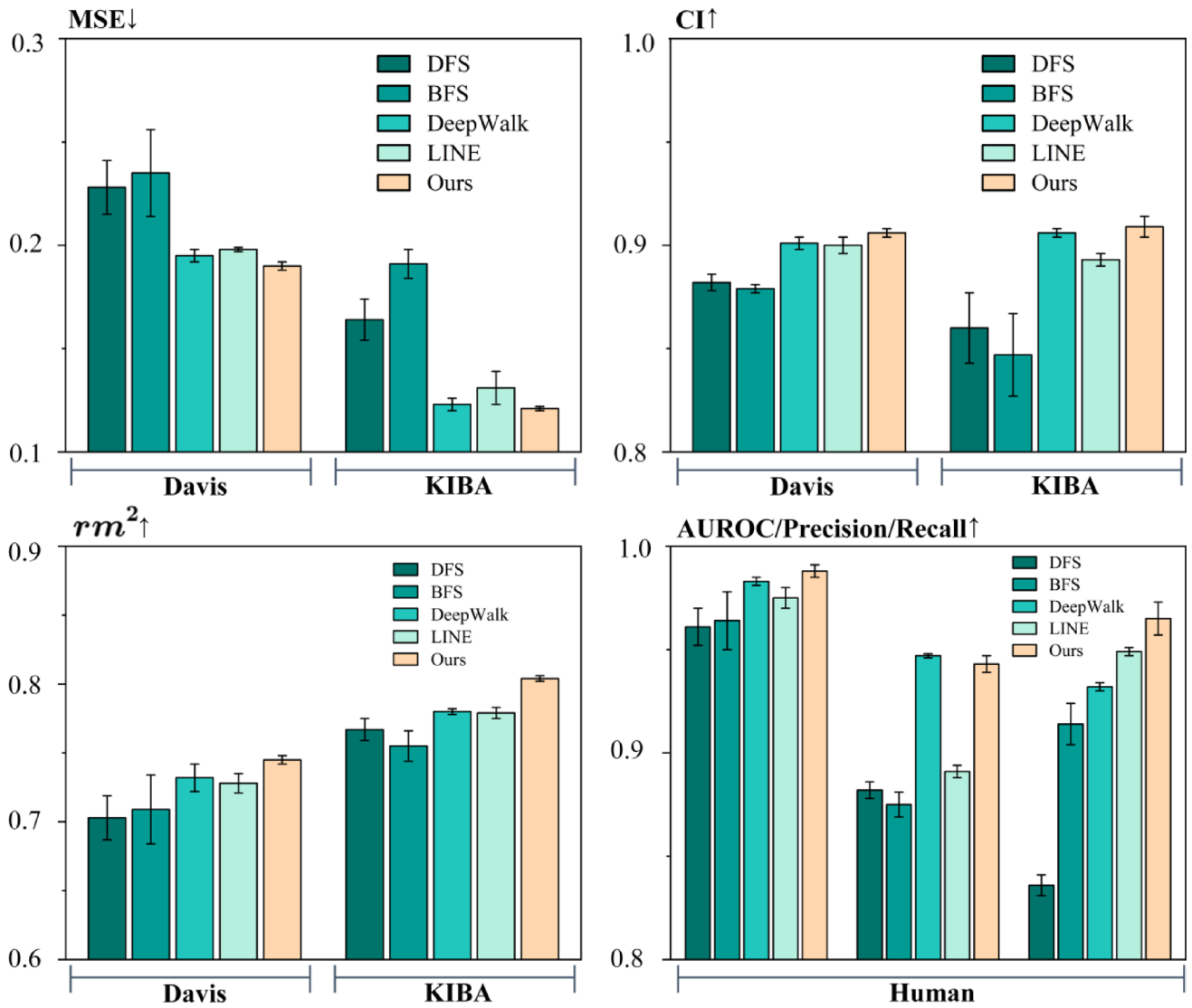

Fig. 4 Performance of different random walk methods and strategies

with MSE of 0.198 and 0.125 on Davis and KIBA, and AUROC of 0.975 on Human.

Performance of different parameter combinations

BFS, DFS, and the balanced sampling strategies are actually three representative conditions of random walk. To get a full glance of the hyperparameter landscape, we further evaluated a series of $<p, q>$ combinations and plot the model performances in Fig. 5.

It can be observed form the landscape that the model performs best when $p=0.25$ and $q=0.5$, while the results are the same and the worst when $p=0.75$ and $q=1$, as well as $p=1$ and $q=0.5$. In general, the overall performance of the model deteriorates with the increase of p and q. We have described in section **Protein representation learning via random walk on the PPI network** that $q < 1$ indicates higher chance of visiting nodes further away from the starting node. This reflects a better aggregation of the neighborhood information of distant nodes, rather than extracting from overly localized views. On the other hand, this remote sampling is not always encouraged without restriction. In this aspect, p controls the possibility of accessing back nodes during node traversal, and a smaller value of p $(< \min(q, 1))$ corresponds to a higher probability of backward step. Clearly, $p=0.25$ suggests a moderate exploration of the remote proportion of the network, which avoids the aggregation of irrelevant noise and the subsequent performance decrease.

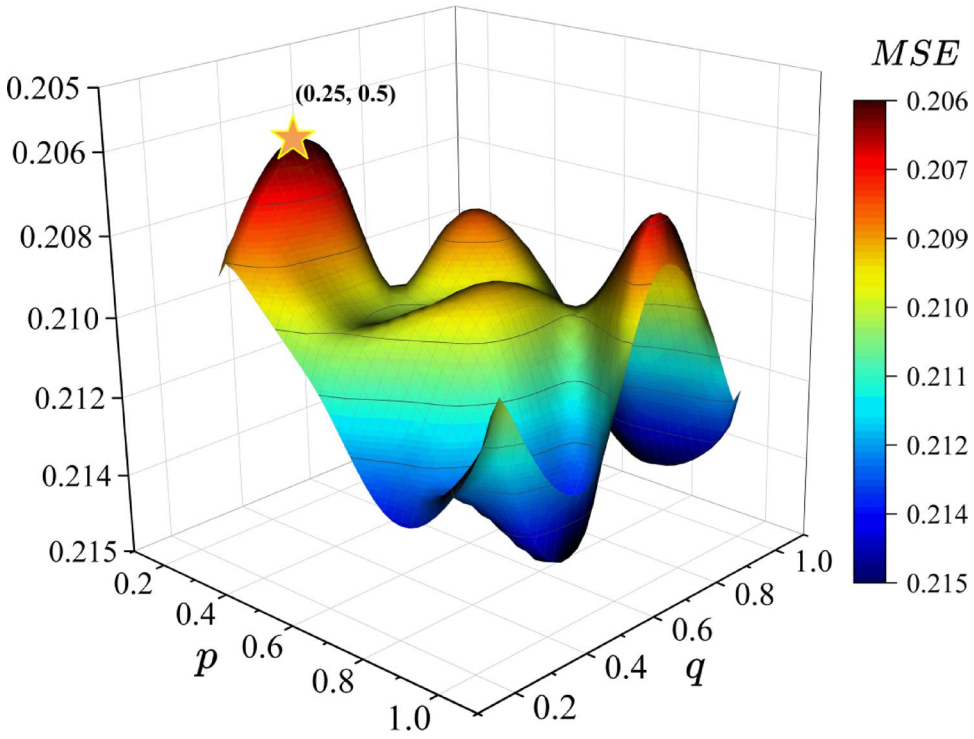

Fig. 5 The landscape of $<p, q>$ hyperparameter combinations

Performance of different knowledge distillation strategies

As disclosed in previous section, LightDTA may adopt three different knowledge distillation strategies, namely the response-based distillation, the joint feature-based distillation, and the protein feature-based distillation. To determine the optimal strategy, we trained three model variants of LightDTA using different distillation strategies and evaluated them accordingly. The experimental results are shown in Fig. 6.

According to the experimental results, the response-based distillation strategy consistently performs the best on all datasets. Specifically, the response-based distillation achieved MSE scores of 0.190 and 0.121 on Davis and KIBA, respectively, and achieved a very high AUROC of xxx on the Human dataset. In contrast, the second varies depending on the task. For example, the joint feature-based strategy achieves the second on the two DTA datasets, while the protein feature-based strategy is in turn better than the joint feature-based one on the CPI task. Nevertheless, the response-based distillation always be the best strategy among three candidates. This is attributed to its ability to effectively extract the teacher model's probabilistic cognition of drug target interaction patterns, especially by fully transmitting the correlations learned by the teacher model to the student model through soft labels and temperature regulation.

As shown in Table 5, the teacher model demonstrates a superior predictive performance on both Davis and KIBA datasets, achieving a better MSE (0.192) compared to the student model (0.196). After the application of knowledge distillation, the distilled student model demonstrated comprehensive improvements. Specifically, on the Davis dataset, it achieved an improved MSE of 0.190 and r_m^2 of 0.745, while on the KIBA dataset, its r_m^2 reached a notable score of 0.804. In brief, the distilled student model outperformed the teacher model across all metrics while maintaining the lightweight advantage of the original student model. A similar trend was observed in the CPI task. As shown in Table 6, the distilled student model achieved a better Recall (0.965), which substantially outperformed the teacher model. Though slightly inferior in terms of the other two metrics, the distilled model maintained a competitive performance.

Comparisons with state-of-the-art approaches in the DTA task

We first evaluated the performance of LightDTA on the two DTA benchmark datasets, and compared it with several state-of-the-art methods. The evaluation results are presented in Table 7. As observed in the table, LightDTA apparently outperforms existing methods despite using a lightweight architecture. Specifically, on the Davis dataset, LightDTA achieves an MSE of 0.190 and a CI of 0.906, which is considerably superior to other lightweight models like

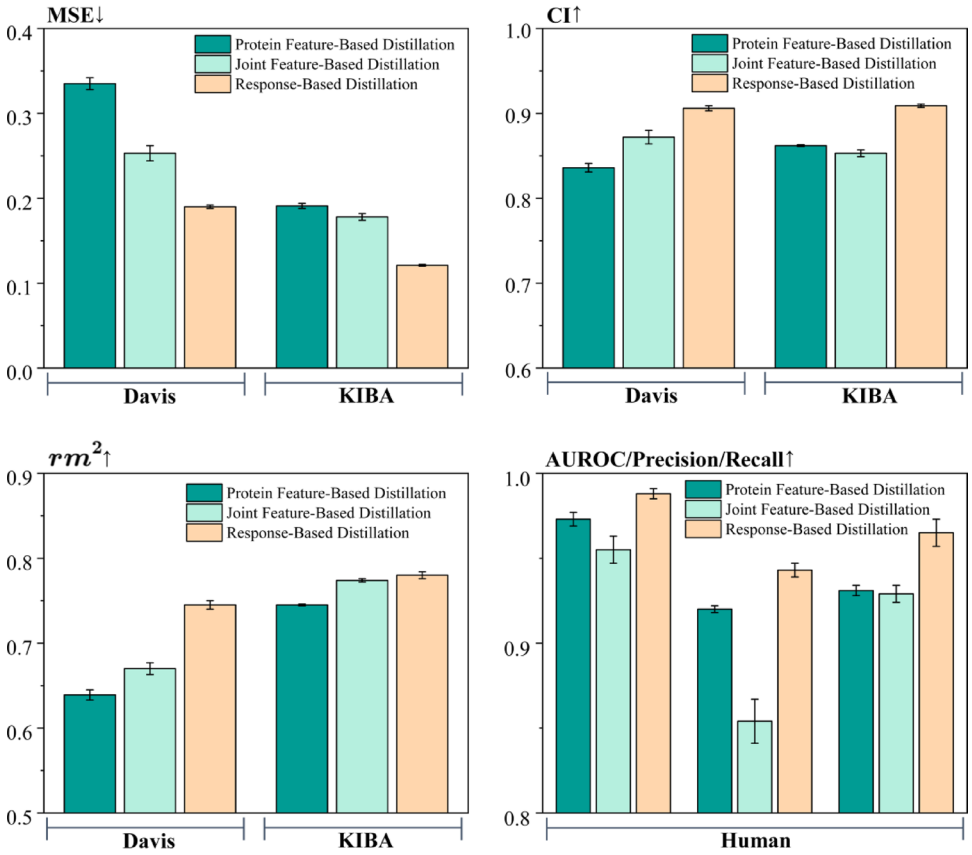

Fig. 6 Performance of different distillation learning methods

Table 5 Performance comparison of different methods on the Davis and KIBA datasets

Dataset	Davis			KIBA		
Method	$MSE\downarrow$	$CI\uparrow$	$r_m^2\uparrow$	$MSE\downarrow$	$CI\uparrow$	$r_m^2\uparrow$
Vanilla Teacher	0.192 (0.001)	0.905 (0.002)	0.731 (0.004)	0.122 (0.002)	0.904 (0.003)	0.793 (0.001)
Vanilla Student	0.196 (0.003)	0.904 (0.004)	0.725 (0.005)	0.124 (0.002)	0.908 (0.002)	0.786 (0.003)
Distilled Student	**0.190 (0.002)**	**0.906 (0.002)**	**0.745 (0.003)**	**0.121 (0.001)**	**0.909 (0.005)**	**0.804 (0.002)**

Optimal results are shown in bold, and sub-optimal results are indicated by underlying

Table 6 Performance comparison of different methods on the Human dataset

Methods	AUROC	Precision	Recall
Vanilla Teacher	**0.988 (0.001)**	**0.945 (0.003)**	0.952 (0.006)
Vanilla Student	0.985 (0.004)	0.942 (0.002)	0.954 (0.005)
Distilled Student	0.988 (0.003)	0.943 (0.004)	**0.965 (0.008)**

Optimal results are shown in bold, and sub-optimal results are indicated by underlying

DeepDTA (MSE=0.255) and WideDTA (MSE=0.261). LightDTA also outperforms GPCNDTA (MSE=0.261) where the model architecture is complicated and is computational expensive upon training. Since LightDTA gains the molecular-level features via distilling from the teacher model, it does not inferior to these molecular-graph based methods. As for the KIBA dataset, LightDTA also achieves state-of-the-art performance when comparing to other methods. Note that the hyperparameters are optimized on Davis and are not overfit to specific dataset; therefore, the above results together prove the generalizability across datasets.

We randomly partitioned the Davis dataset into ten distinct training-test set pairs and used the MSE metric to assess significance. First, we employed the Shapiro–Wilk test to confirm that the results did not deviate from a normal distribution, followed by the paired Student's t-test to examine whether the differences between the results were statistically significant. As shown in Fig. 7, the differences between LightDTA and other baseline methods are statistically significant. Specifically, LightDTA achieved a significance level of $p < 0.05$ compared to the second-best method, and $p < 0.01$ compared to all other methods.

Furthermore, we supplemented the evaluation of the model on Metz dataset, which has 1,423 compounds, 170 proteins and 35,259 interactions [49]. The affinity values are reported using pIC50, thus forming a regression task. As shown in Table 8, LightDTA achieves consistently best

Table 7 Performance comparison of different methods on the Davis and KIBA datasets

Dataset	Davis			KIBA		
Method	$MSE\downarrow$	$CI\uparrow$	$r_m^2\uparrow$	$MSE\downarrow$	$CI\uparrow$	$r_m^2\uparrow$
DeepDTA [5]	0.255 (0.003)	0.874 (0.002)	0.703 (0.005)	0.188 (0.001)	0.851 (0.004)	0.760 (0.002)
WideDTA [48]	0.261 (0.004)	0.884 (0.002)	0.711 (0.003)	0.251 (0.006)	0.880 (0.003)	0.765 (0.004)
GraphDTA [9]	0.229 (0.002)	0.894 (0.004)	0.719 (0.001)	0.144 (0.003)	0.887 (0.002)	0.771 (0.001)
DGraphDTA [10]	0.203 (0.003)	0.905 (0.002)	0.721 (0.002)	0.126 (0.001)	0.907 (0.004)	0.780 (0.001)
MGraphDTA [12]	0.208 (0.006)	0.899 (0.002)	0.720 (0.005)	0.128 (0.002)	0.901 (0.001)	0.796 (0.004)
GPCNDTA [30]	0.197 (0.006)	0.903 (0.004)	0.723 (0.006)	0.121 (0.002)	0.907 (0.003)	0.785 (0.003)
LightDTA(Ours)	**0.190 (0.002)**	**0.906 (0.002)**	**0.745 (0.003)**	**0.121 (0.001)**	**0.909 (0.005)**	**0.804 (0.002)**

Optial results are shown in bold, and sub-optimal results are indicated by underlying

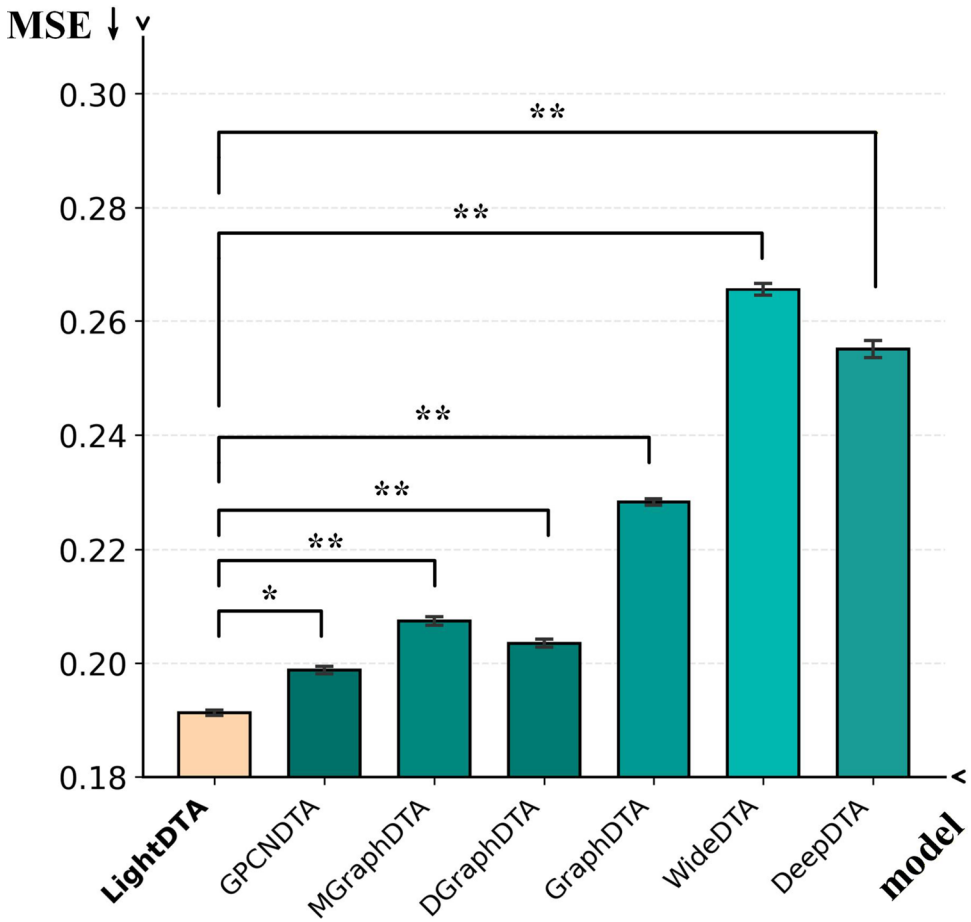

Fig. 7 Statistical significance analysis of model performance

Table 8 Performance of LightDTA on Metz dataset

Methods	$MSE\downarrow$	$CI\uparrow$	$r_m^2\uparrow$
DeepDTA	0.286 (0.001)	0.815 (0.001)	0.678 (0.003)
WideDTA	0.313 (0.003)	0.800 (0.002)	0.641 (0.007)
GraphDTA	0.282 (0.011)	0.816 (0.004)	0.681 (0.026)
DGraphDTA	0.254 (0.003)	0.825 (0.002)	0.709 (0.004)
MGraphDTA	0.265 (0.002)	0.822 (0.001)	0.701 (0.001)
GPCNDTA	0.251 (0.010)	0.826 (0.001)	0.698 (0.009)
LightDTA (Ours)	**0.223 (0.001)**	**0.847 (0.002)**	**0.728 (0.002)**

Optimal results are shown in bold, and sub-optimal results are indicated by underlying

performance across all metrics. Compared to the second-best results, it reduces MSE by 2.8%, improves CI and enhances r_m^2 by 2.1% and 1.9%, respectively.

To be more expressive, the experimental and predicted binding affinities for Davis and KIBA datasets are organized using a correlation plot (Fig. 8) based on over 3,000 pairs of samples. An idealized model would generate a linear correlation where the predicted values perfectly equaling to the ground truth. It can be observed that the predicted affinities by LightDTA are highly correlated with the experimentally measured values, suggesting a good predictive capacity of the proposed model.

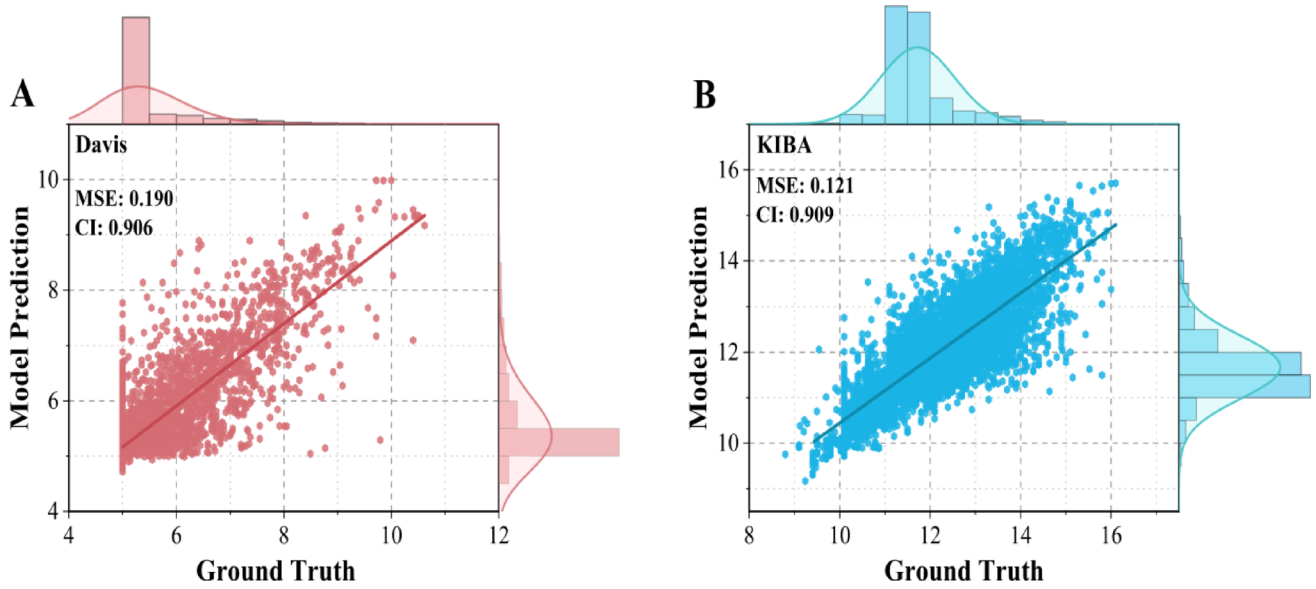

Fig. 8 Correlation plot for **(A)** Davis and **(B)** KIBA datasets given by LightDTA

Table 9 Performance comparison of different methods on the Human dataset

Methods	AUROC	Precision	Recall
RF [50]	0.944 (0.010)	0.899 (0.004)	0.866 (0.006)
GCN [50]	0.957 (0.002)	0.864 (0.003)	0.928 (0.001)
TransformerCPI [50]	0.972 (0.003)	0.911 (0.007)	0.922 (0.008)
GraphDTA [9]	0.973 (0.002)	0.883 (0.009)	0.914 (0.020)
MGraphDTA [12]	0.984 (0.003)	**0.952 (0.006)**	0.955 (0.005)
LightDTA (Ours)	**0.988 (0.003)**	0.943 (0.004)	**0.965 (0.008)**

Optimal results are shown in bold, and sub-optimal results are indicated by underlying

Table 10 Performance of LightDTA and baseline models on the Davis dataset under the cold-pair scenario

Methods	$MSE \downarrow$	$CI \uparrow$	$r_m^2 \uparrow$
DeepDTA	0.879 (0.023)	0.553 (0.019)	0.582 (0.007)
WideDTA	0.893 (0.011)	0.547 (0.005)	0.604 (0.010)
GraphDTA	0.826 (0.014)	0.563 (0.013)	0.628 (0.009)
DGraphDTA	0.808 (0.012)	0.591 (0.003)	0.623 (0.015)
MGraphDTA	0.782 (0.002)	0.615 (0.017)	0.644 (0.011)
GPCNDTA	0.785 (0.004)	0.619 (0.003)	0.630 (0.009)
LightDTA (Ours)	**0.767 (0.007)**	**0.634 (0.011)**	**0.688 (0.006)**

Optimal results are shown in bold, and sub-optimal results are indicated by underlying

Comparisons with state-of-the-art approaches in the CPI task

To further demonstrate the cross-task generalization ability of LightDTA, we evaluated it using the Human dataset, which is a CPI dataset of binary classification. The performance of LightDTA, along with the baseline methods are shown in Table 9. It can be observed that LightDTA achieved the best performance in terms of AUROC (0.988) and Recall (0.965), and the second performance in terms of Precision (0.943), suggesting a good generalizability of LightDTA across tasks.

Performance evaluation under cold-start scenarios

During the aforementioned training process, the training-test splits were all randomly partitioned rather than adopting temporal or structural-based schemes, which maintains consistency with established benchmarks in this field and facilitates fair comparisons with previous baseline models. However, the above random splitting scheme does not reflect the real-world conditions. Therefore, to systematically evaluate the model's generalization capability and performance under more challenging scenario, we employed two cold-start settings by adopting temporal and structural data splitting strategies, respectively. In addition, we also test the model on an external COVID-DTA dataset that aligns with the real-world applications.

Cold-start with temporal-based splitting strategy

We implemented this approach by masking proteins (and their interactions) that appeared exclusively in the test set when constructing the PPI network. Concurrently, we re-partitioned the dataset to ensure complete exclusion of any drugs or proteins from the test set during the training process. Based on the reconstructed PPI network and re-partitioned dataset, we re-evaluated LightDTA and all baseline models. The evaluation results are presented in Table 10.

According to the evaluation results, the proposed LightDTA achieved SOTA performance across all evaluation

metrics, with the lowest MSE of 0.767, the highest CI of 0.634, and r_m^2 of 0.688 among all the compared models. The good performance under the challenging cold-start scenario suggests that LightDTA is able to generalize well to real-world drug discovery.

Cold-start with scaffold-based splitting strategy

To assess the model performance in terms of the structural splitting view, we adopted a scaffold-based partitioning approach and conducted a more rigorous assessment. According to the experimental results, LightDTA still achieved the best performance among all baseline models, with an MSE of 0.588 that notably better than the suboptimal model DGraphDTA (MSE = 0.624). In addition, LightDTA performs consistently well in terms of other metrics. These together demonstrate the good generalizability of the proposed model to novel chemical scaffolds. The evaluation results are shown in Table 11.

Cold-start on external COVID-DTA dataset

To further demonstrate the generalization of the proposed model on out-of-distribution and out-of-domain samples, we conducted another external test set, i.e., COVID-DTA, which is collected by Chen et al. [51]. Specifically, we fetched available data from the repository of COVID-DTA and obtained 1,852 drug-target pairs with 5 target proteins. The affinity values are reported using pIC50, thus forming a regression task. Detailed statistics are shown in Table 12.

The evaluation results of LightDTA and baseline models are shown in Table 13. Accordingly, LightDTA achieved the optimal MSE of 0.426 among all tested models, while obtaining suboptimal results in terms of CI and r_m^2 metrics. Although the performance scores of the latter two metrics are slightly lower than that of MGraphDTA, they generally remain competitive.

Performance of alternative PPI embedding methods

The random walk-based protein node embeddings in this study is motivated by biological intuitions. It is designed to capture the functional relationships within the protein–protein interaction (PPI) network, which is well-established in biological literatures as a meaningful representation of protein functional similarities. Furthermore, the initial embedding via random walk is able to reflect both local links (second- and third-order neighboring nodes) and more distant (higher-order) topological relationships. This is of particular importance for proteins since lots of them exert functions not only by collaborating directly with near proteins, but also by involving in a pathway so that to collaborate indirectly with more distant proteins. In contrast, other graph embedding methods like graph neural networks (GCN) and graph autoencoders (GAE) are often limited by depth and therefore tend to capture only low-order neighbors. To practically demonstrate the advantage of random walk in robustness over other methods, we implemented three alternative embedding approaches including GCN, GAE, and GAT (i.e., graph attention network), and assessed their performance on the Davis dataset, as reported in Table 14.

Table 11 Frequency based scaffold partitioning approach (five random runs)

Methods	$MSE \downarrow$	$CI \uparrow$	$r_m^2 \uparrow$
DeepDTA	0.840 (0.006)	0.675 (0.013)	0.607 (0.008)
WideDTA	0.837 (0.015)	0.679 (0.008)	0.605 (0.012)
GraphDTA	0.818 (0.002)	0.685 (0.001)	0.718 (0.004)
DGraphDTA	0.624 (0.023)	0.733 (0.014)	0.619 (0.010)
MGraphDTA	0.678 (0.017)	0.731 (0.005)	0.629 (0.015)
GPCNDTA	0.680 (0.019)	0.738 (0.015)	0.726 (0.008)
LightDTA (Ours)	**0.588 (0.020)**	**0.765 (0.011)**	**0.743 (0.006)**

Optimal results are shown in bold, and sub-optimal results are indicated by underlying

Table 12 The details of the COVID-DTA dataset

Name	Organism	Uniprot	PDB	DT pairs
ACE2	Human	Q9BYF1	7U0N	122
TMPRSS2	Human	O15393	7MEQ	18
pp1ab	SARS2	P0DTD1	6Z5T	1636
RdRp	SARS	Q70GC6, O39930	–	76

Optimal results are shown in bold, and sub-optimal results are indicated by underlying

Table 13 Performance of COVID dataset

Methods	$MSE \downarrow$	$CI \uparrow$	$r_m^2 \uparrow$
DeepDTA	0.715 (0.016)	0.678 (0.011)	0.674 (0.015)
WideDTA	0.717 (0.013)	0.676 (0.012)	0.660 (0.008)
GraphDTA	0.632 (0.011)	0.715 (0.010)	0.701 (0.014)
DGraphDTA	0.487 (0.007)	0.783 (0.013)	0.706 (0.011)
MGraphDTA	0.445 (0.012)	**0.813 (0.005)**	**0.718 (0.009)**
GPCNDTA	0.468 (0.016)	0.785 (0.014)	0.709 (0.007)
LightDTA (Ours)	**0.426 (0.008)**	0.791 (0.006)	0.715 (0.004)

Optimal results are shown in bold, and sub-optimal results are indicated by underlying

Table 14 Performance of LightDTA variants with different embedding methods

Methods	$MSE \downarrow$	$CI \uparrow$	$r_m^2 \uparrow$
GCN	0.214 (0.001)	0.901 (0.002)	0.708 (0.014)
GAT	0.212 (0.002)	0.902 (0.003)	0.713 (0.003)
GAE	0.216 (0.004)	0.900 (0.003)	0.696 (0.001)
Ours	**0.190 (0.002)**	**0.906 (0.002)**	**0.745 (0.003)**

Optimal results are shown in bold, and sub-optimal results are indicated by underlying

In addition, to compare with the path-based sampling method such as Random Walk [52], we also used the classical neighborhood-based sampling approach GraphSAGE [53], as well as the subgraph- based sampling method Cluster-GCN [54]. As shown in Table 15, path-sampling method generally performed better than other strategies, and our approach (the tuned biased random walk strategy) achieved the best. Although DeepWalk achieved stable performance as a sub optimal result, the randomly generated sequences it samples struggle to capture the balance between the cohesiveness of local functional clusters and long range functional associations in PPI network. As a representative subgraph-sampling strategy, Cluster-GCN pre-partitions the entire PPI network into multiple dense sub-graphs. While this strategy can effectively aggregate protein functional associations within each cluster and improve computational efficiency during training, its restricted sampling scope may sever important cross-cluster functional linkages, thereby limiting the model's ability to learn the semantics of the global functional network. The underperforming GraphSAGE method samples a fixed number of nodes randomly from the neighbors of each target protein. This random, fixed-size neighborhood sampling mechanism cannot establish consistent long range receptive fields. So the lack of global semantics limits its performance, ultimately leading to a decline in predictive accuracy.

Table 15 Performance of LightDTA variants with different sampling strategies

Method	Sampling strategy	$MSE \downarrow$	$CI \uparrow$	$r_m^2 \uparrow$
GraphSAGE	Neighborhood-based	0.501 (0.001)	0.805 (0.001)	0.368 (0.003)
Cluster-GCN	Subgraph- based	0.277 (0.004)	0.877 (0.002)	0.640 (0.009)
Deep Walk	Path- based	<u>0.198</u> <u>(0.001)</u>	<u>0.900</u> <u>(0.002)</u>	<u>0.730</u> <u>(0.002)</u>
Ours		**0.190 (0.002)**	**0.906 (0.002)**	**0.745 (0.003)**

Optimal results are shown in bold, and sub-optimal results are indicated by underlying

Table 16 Compound ranking based on Light DTA prediction of affinity targeting Prostaglandin G/H synthase 2 receptor

Approved Drug ID	Name	Ranking
CHEMBL122	ROFECOXIB	5 (Top 5%)
CHEMBL1316	CARPROFEN	19 (Top 5%)
CHEMBL1070	NABUMETONE	73 (Top 5%)
CHEMBL118	CELECOXIB	126 (Top 5%)
CHEMBL865	VALDECOXIB	215 (Top 10%)
CHEMBL622	ETODOLAC	287 (Top 15%)
CHEMBL416146	ETORICOXIB	292 (Top 15%)
CHEMBL 1206690	PARECOXIB	360 (Top 15%)
CHEMBL599	MELOXICAM	441 (Top 20%)
CHEMBL404108	LUMIRACOXIB	458 (Top 20%)

Case analysis on drug screening in real-world scenarios

For the cross-domain generalization, we used the single protein Prostaglandin G/H synthase 2 (Uniprot_ID: P35354) together with approved drugs from DrugBank to evaluate the model performance. To simulate a realistic screening scenario, LightDTA was trained on the Davis dataset and then used to predict the binding affinity between P35354 and 2,648 potential small molecules. It should be noted that P35354 does not appear in the Davis dataset, ensuring that our setup reflects a real-world drug screening process. Among the 2,648 compounds, 10 are known P35354-targeting drugs retrieved from ChEMBL. The predicted affinity rankings of these 10 drugs against P35354 are shown in Table 16. The results demonstrate that all 10 approved drugs are ranked within the top 18% of all candidates, with 4 of them appearing in the top 5%. These findings indicate that LightDTA can effectively transfer the generalized drug-target interaction patterns learned from the Davis dataset to the affinity ranking task for a novel target, P35354, demonstrating strong cross-domain generalization capability.

Case analysis on binding pocket awareness via knowledge distillation

The representation learning of the student model for PPI networks does not explicitly rely on traditional biochemical attributes or protein structural information. Therefore, the implicit supervisory signals provided by the teacher model through knowledge distillation, which integrate structural semantics and biochemical constraints, are crucial for enhancing the interpretability of the student model.

The interaction between a drug and its target usually does not involve the entire protein, but is highly specific to certain functional regions on its surface, namely binding pockets. Therefore, the ability of a model to accurately capture information related to protein pockets is crucial for predicting drug–target interactions, and also provides a reasonable basis for evaluating model quality and interpretability. To examine the capacity of the teacher model to learn protein pocket information, we conducted a visualization analysis based on Gradient-weighted Activation Mapping (Grad-AAM) [55]. Specifically, we first used the trained model to predict the affinity scores of drug–target pairs, and then performed backpropagation on these scores to compute the gradients of the features output by the structural graph convolutional encoder. These gradients reflect the contribution

of different residues to the final affinity prediction. The visualization results are shown in Fig. 9.

It can be observed that the teacher model can captured the specific pocket regions, demonstrating a good interpretability in combination with Grad-AMM.

Case analysis on the sufficiency of the lightweight protein representation

The sufficiency of the lightweight protein representation is attributed to its capability to preserve essential functional and contextual information directly derived from the PPI network topology. By performing random walks on PPI network, we obtain a low-dimensional vector that effectively captures a protein's positional information within biological contexts and functional modules. This approach essentially converts network topology into sequences, enabling functionally similar or related proteins to be mapped to similar vector representations. Meanwhile, by controlling walk length, the method captures not only local and highly correlated functional clusters but also indirect protein associations organized around biological pathways.

As shown in Fig. 10, we visualized the protein–protein interaction (PPI) network constructed from 170 proteins in the Metz dataset using STRING and performed module identification using two clustering approaches: the density-based method DBSCAN and the random walk-based Markov Clustering Algorithm (MCL).

In density-based clustering (DBSCAN), proteins are grouped into clusters according to their proximity in the feature space, while nodes in sparse regions are labeled as noise. In PPI networks, which are often sparsely and unevenly connected, this method tends to struggle with effectively identifying functional modules characterized by complex topological relationships. As a result, many genuine interactions may be misclassified as noise, leading to the loss of critical biological information. In contrast, the random walk-based clustering operates directly on the network topology by simulating random walks. Through iterative enhancement and weakening of inter-cluster connections based on walk probabilities, it naturally partitions the network into subnetworks. This approach not only effectively identifies locally strong protein–protein interactions but also captures long-range dependencies spanning multiple nodes.

Case analysis on critical targets driving malignant tumors

To showcase the practical application of LightDTA, we conducted a case analysis on a critical target known to drive malignant tumors. Specifically, the epidermal growth factor receptor (EGFR), as an important member of the ErbB receptor tyrosine kinase family, plays a central regulatory role in cell proliferation, differentiation, survival, and tumorigenesis through its signaling cascade. Existing research has identified the key driving role of EGFR in malignant tumors such as non-small cell lung cancer (NSCLC) and colorectal cancer (CRC), while revealing its pathological mechanisms in non-tumor diseases such as pulmonary fibrosis and psoriasis. We adopt LightDTA to screen for high affinity ligands targeting EGFR from Davis dataset. Accordingly, two marketed drugs known to target EDFR receptors, i.e., Gefitinib and Osimertinib, were successfully identified with high predicted pKd affinities of 9.84 and 10.15, respectively. In addition, LightDTA also screened out other two lead compounds targeting EDFR, showing highly consistent pKd with experimental measurements ($\Delta \leq 0.5$).

The efficiency of LightDTA helps to achieve a fast initial screening. This is followed by a more refined molecular

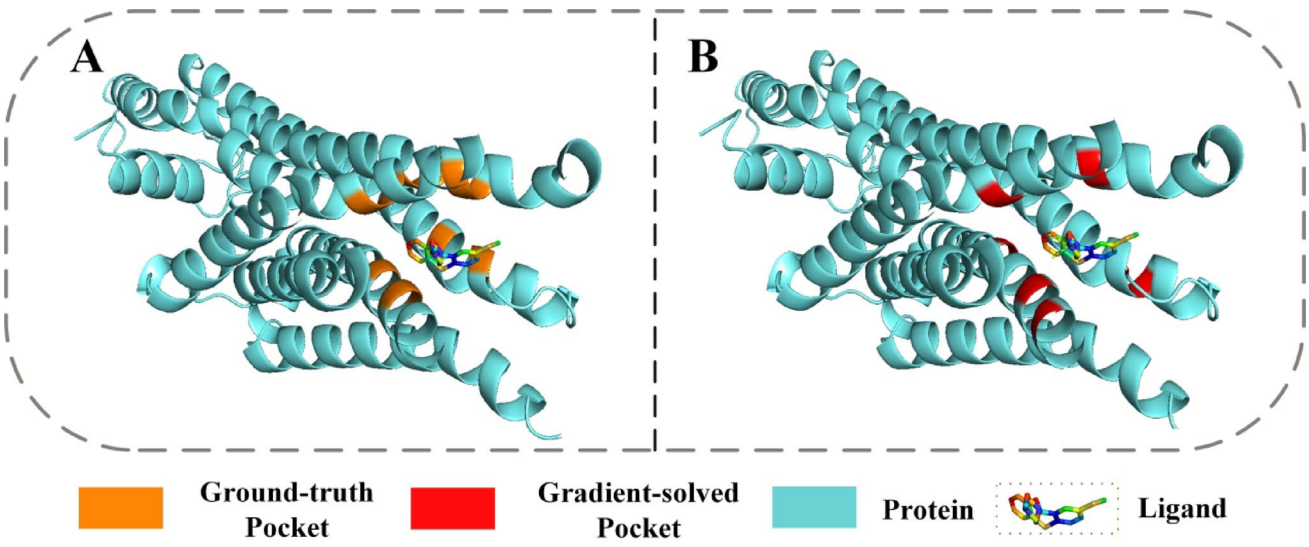

Fig. 9 Visualization of the predicted binding pockets by the teacher model

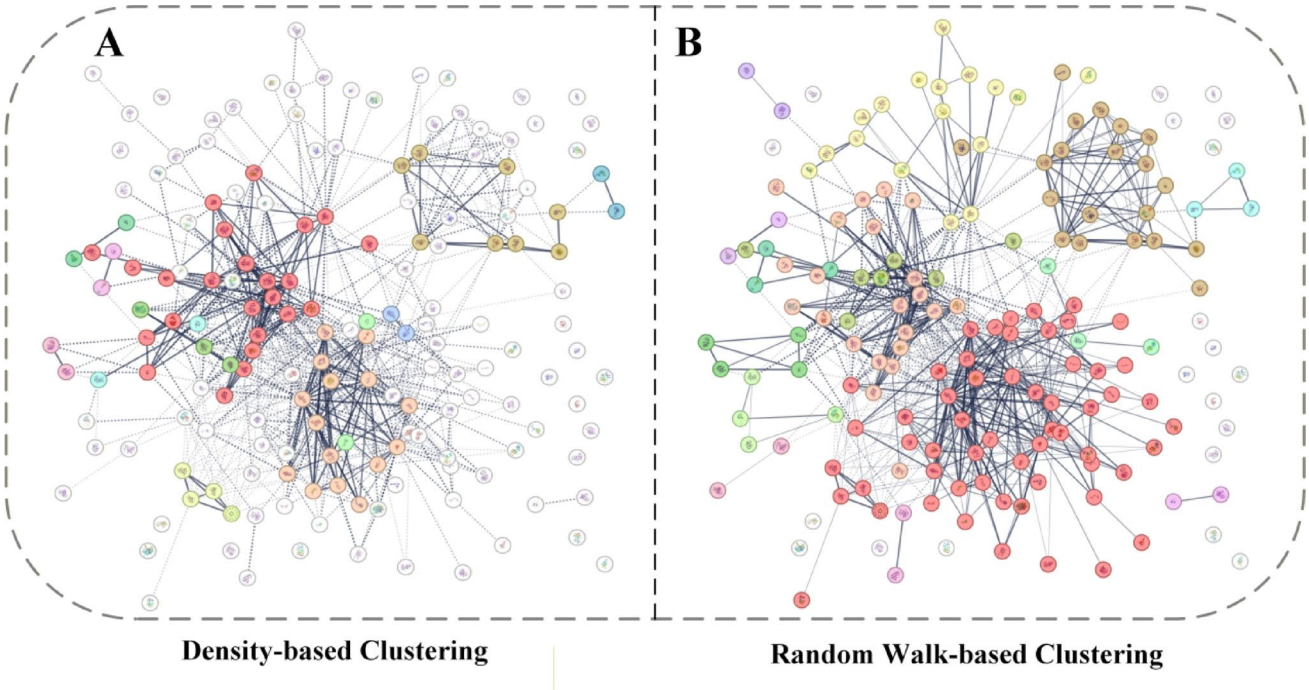

Fig. 10 Different clustering methods on PPI networks

docking for a further validation. Here we analyzed the above two lead compounds by performing molecular docking in Autodock Vina and visualizing the docking results, as shown in Fig. 11. According to the docking results, each of the two molecules formed more than three high-affinity hydrogen bonds with EGFR. In terms of the docking score, the two molecules achieved binding energies of − 8.4 and − 7.9 kcal/mol, respectively. Typically, a binding affinity Kc≤− 5.0 suggests a plausible conformation with detectable interaction, while Kc≤− 7.0 signifies strong binding potential, marking the compound as a promising drug candidate. Therefore, the docking scores align well with the predicted affinities given by LightDTA, which further validate the predictive performance of the proposed model.

Analysis of computational complexity

To quantitatively demonstrate the lightweight nature of LightDTA, we compared its computational cost with that of other baseline models, as shown in Fig. 12. It can be observed that LightDTA not only achieves state-of-the-art performance but also demonstrates excellent efficiency in terms of both resource requirement and inference time. Particularly, training LightDTA requires only 3978 MB (megabytes) of GPU memory, which is much smaller than the 6485 MB required by the second-best model GPCNDTA. Other baseline methods with similar training requirements, such as DeepDTA and GraphDTA released in earlier years, are significantly inferior to the proposed LightDTA in predictive performance.

In addition, LightDTA also demonstrates advantages in inferring. LightDTA takes only 1.9 s for the test set of Davis (the specific running time depends on the test platform). In contrast, GPCNDTA took 15 s to complete 5010 samples, 7-times longer than LightDTA and achieved only suboptimal results. The inference time is particularly important for practical employment where large-scale molecules are to be screened with limited hardware resources, and the proposed method provides a more feasible and efficient solution for real-world applications.

Discussion

The advantage of random walk algorithm

Before diving deeper into the role of random walk algorithm, we would first like to briefly introduce the construction of the protein–protein interaction (PPI) network and the knowledge embedded within its topology. Actually, STRING has aggregated abundant knowledge of proteins from UniProt, KEGG, NCBI, and Gene Ontology. Interaction edges by STRING can reflect multiple evidence types including neighborhood, fusion, co-occurrence, experiments, co-expression, databases of pathway and compound, and text mining from high-confidence literature, etc. [56].

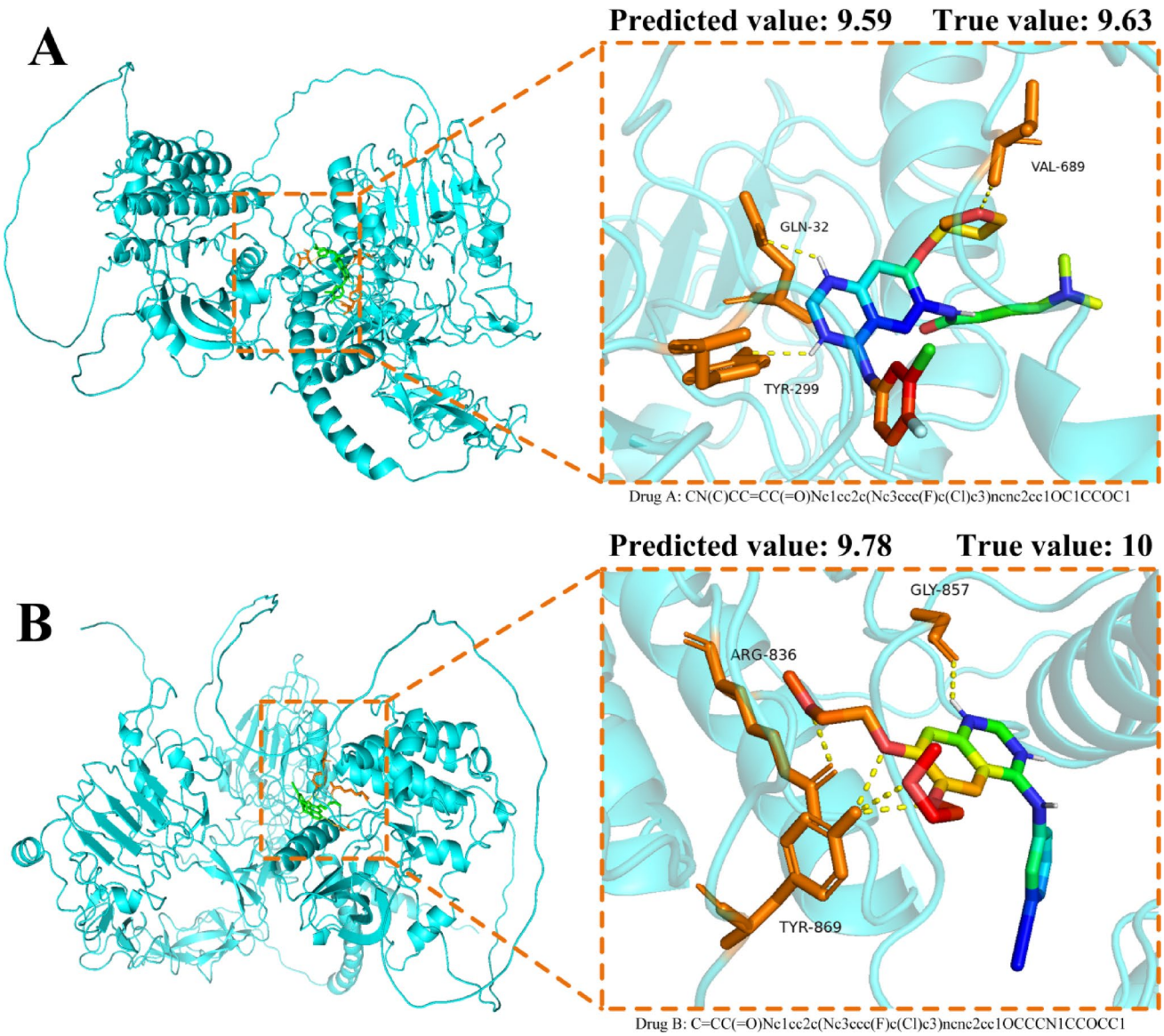

Fig. 11 Representative high affinity candidate drugs targeting EFGR receptors

These together prepare a good knowledge base for the random walk algorithm.

In our model, the random walk algorithm samples a sequence of nodes from this network, converting the graph structure into learnable sequential data. It naturally capture the proximity patterns among proteins in the functional network. Therefore, proteins that frequently co-occur in the walk sequences or topologically close in the network are mapped to nearby positions in the vector space during embedding learning. Because the PPI network inherently reflects functional associations and similarities, the embeddings learned from the walk sequences naturally cluster functionally related or similar proteins. Through this sampling and embedding process, the low-dimensional vector representations encode functional associations and topological semantics, providing essential features for the subsequent prediction task. Moreover, the random walk strategy offers flexible tunability. By adjusting walk parameters such as walk length, return parameter p, and in-out parameter q, the model can balance the exploration of local structure against the capture of global relationships. This biased random walk design enables us to flexibly balance the cohesiveness of local functional clusters and the connectivity of the global functional network, thereby allowing a more comprehensive learning of the biological information of proteins within the complex interaction network.

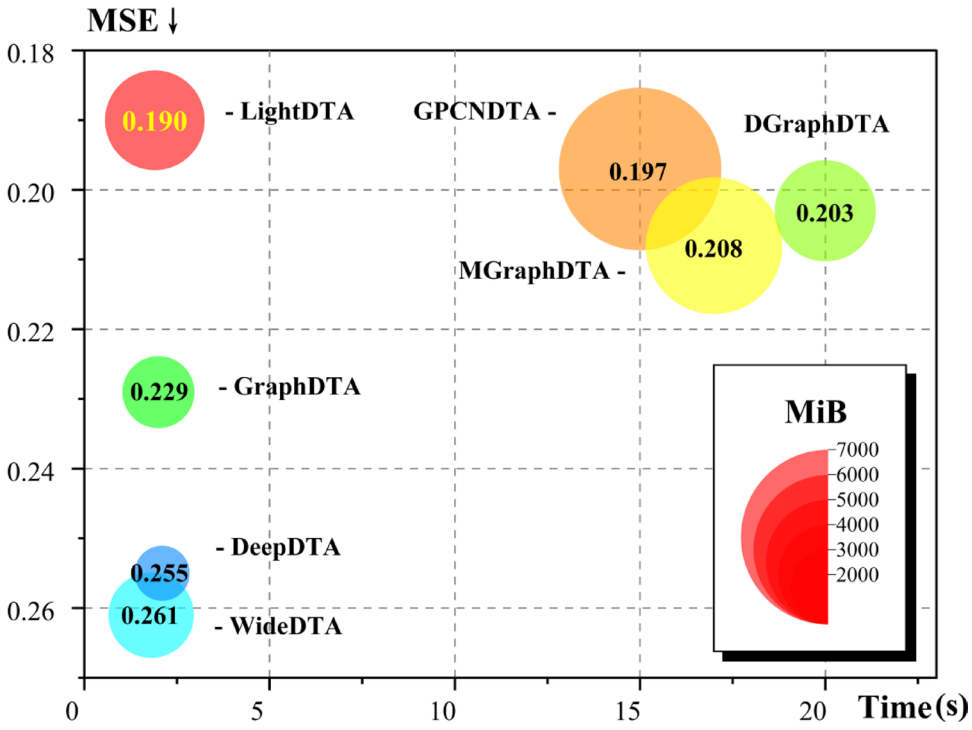

Fig. 12 Comparison of different methods in terms of resource requirement of GPU memory and inference time

Limitation

In reviewing existing models, we observed that their high computational resource requirements often constitute a bottleneck for screening large-scale compound libraries in real-world scenarios, thereby limiting their practical deployment. Based on this observation, LightDTA was designed with computational efficiency as priority, aiming to significantly reduce the computational burden during prediction through a lightweight architecture, thereby enhancing the model's scalability and practicality in large-scale virtual screening tasks.

However, as the reviewer mentioned, the emphasis on efficiency and simplicity inevitably entails a trade-off in terms of biochemical interpretability. Despite that the lightweight model is still able to learn biochemical and structural information from teachers via distillation, this process result in the loss of interpretability implicit in the teacher model, particularly those related to specific biophysical mechanisms. Furthermore, the student model's design of PPI network further accentuates this trade-off. Its architecture prioritizes learning from high-level relational patterns within the protein network over low-level physicochemical properties. While this abstraction significantly reduces the computational burden, it comes at the expense of direct interpretability at the biochemical feature level.

Therefore, LightDTA gains efficiency at the expense of fine-grained biochemical interpretability. This design choice is motivated by the pressing need for speed and resource efficiency in large-scale preliminary screening, with the goal of rapidly narrowing down candidate sets to provide high-quality subsets for subsequent experiment.

Future work

In future work, on one hand, the lightweight design philosophy and strategies like knowledge distillation employed in this model can be transferred to other computational biology tasks, further enhancing the efficiency and practicality of related methods. On the other hand, advanced methodologies from other bioinformatics domains can be actively incorporated to address existing gaps in current drug-target affinity prediction.

For example, in cardiovascular disease diagnosis, the CardioBERT [57] model integrates residue-level contact map prediction with the BERT language model, optimizing feature extraction and enriching the feature fusion process, thereby significantly enhancing the model's interpretability and biochemical relevance. Meanwhile, in the field of liver disease prediction, Almusallam et al. proposed a novel predictive framework [58], which combines feature ranking and projection algorithms to achieve high accuracy. By employing SHapley Additive exPlanations (SHAP) to identify the most important predictive features, this approach enhances model interpretability and compensates for the limitations observed in our current model.

Furthermore, with technological advances, the scale of biological datasets continues to grow exponentially. Distributed and cluster computing platforms have become essential for large-scale data analysis and are increasingly integral to

computational biology. For instance, the Sprak-Pi-DNN [59] computational model proposed by Noor et al. utilizes parallel deep neural networks to accelerate the analysis of large-scale RNA sequences without compromising classification accuracy. This lightweight parallelized architecture designed for ultra-large-scale data offers a viable technical direction for the future development of our model.

Conclusion

In this study, we present a new lightweight approach named LightDTA for drug-target affinity prediction. It is motivated by the practical demands of real-world applications. Specifically, to reduce the burden of collecting intact molecular attributes of proteins, we have constructed a protein–protein interaction network and employed random-walk strategy to learn an efficient protein representation. This representation learning method relies solely on the topological structure of macroscopic topology among proteins, thus circumventing the tedious process of collecting detailed biochemical attributes and increasing the generalizability of the model. Evaluations showed that this lightweight architecture achieved MSE of 0.196 and 0.124 on Davis and KIBA, respectively, which was among the top tier of existing cutting-edge methods. To further enrich the molecular-level knowledge while affecting the lightweight character of LightDTA, we introduce a knowledge distillation framework to transfer rich representations from a comprehensive but computational expensive teacher model to the lightweight model. Multiple distillation strategies have been assessed, and the response-based one is proved to be the best choice. This design contributes to an impressive performance, with MSE of 0.190 and 0.121 on Davis and KIBA, respectively. More importantly, computational analysis showed that the performance was achieved with only 61% of the memory requirement of the suboptimal baseline model, and the inference was 30 times faster than the counterpart. In summary, LightDTA offers a highly efficient and accurate model for real-world employment of DTA prediction.

Author contributions S.Z. and X.B. conceived the idea of this study. X.H. and X.B. designed the methodology and conducted the experiments. X.H. and X.B. implemented the computer code and supporting algorithms. H.J., N.X., and W.M. validated the experimental results. X.H., W.L., Q.C., and F.Y. applied statistical analysis and computational techniques. W.Z. and W.L. provided computing resources. X.H. and S.Z. prepared the visualization and presentation of the experimental results. X.H. and S.Z. wrote the original draft. All authors reviewed the manuscript.

Funding This work was supported by the National Natural Science Foundation of China (NO.62306293), the Natural Science Foundation of Shandong Province (NO.ZR2025MS1069), the Fundamental Research Funds for the Central Universities (NO.202561013), and the 111 Project (NO.B23038).

Data availability All codes are available on GitHub: https://github.com/Huang-zilin/LightDTA-final. The datasets used in this study are publicly available. The Davis and KIBA datasets can be obtained from their respective original sources (https://github.com/hkmztrk/DeepDTA/tree/master/data). The Human dataset is available from the cited publications (https://github.com/masashitsubaki/CPI_prediction). The COVID dataset is available from the cited publications (https://github.com/gxCaesar/GINCM-DTA/tree/main/data). The Metz dataset can be obtained from its original source (https://github.com/simonfqy/PADME).

Declarations

Competing interests On behalf of all authors, the corresponding author states that there is no conflict of interest.

Ethics approval and consent to participate Not applicable.

Consent for publication Not applicable.

References

1. Prasad V, Mailankody S (2017) Research and development spending to bring a single cancer drug to market and revenues after approval. JAMA Intern Med 177:1569–1575. https://doi.org/10.1001/jamainternmed.2017.3601
2. Goozner M (2017) A much-needed corrective on drug development costs. JAMA Intern Med 177:1575–1576. https://doi.org/10.1001/jamainternmed.2017.4997
3. Ranjan A, Bess A, Alvin C, Mukhopadhyay S (2024) MDF-DTA: a multi-dimensional fusion approach for drug-target binding affinity prediction. J Chem Inf Model 64(13):4980–4990. https://doi.org/10.1021/acs.jcim.4c00310
4. Forli S, Huey R, Pique ME et al (2016) Computational protein-ligand docking and virtual drug screening with the AutoDock suite. Nat Protoc 11:905–919. https://doi.org/10.1038/nprot.2016.051
5. Öztürk H, Özgür A, Ozkirimli E (2018) DeepDTA: deep drug–target binding affinity prediction. Bioinformatics 34:i821–i829. https://doi.org/10.1093/bioinformatics/bty593
6. Öztürk H, Ozkirimli E, Özgür A (2019) WideDTA: prediction of drug-target binding affinity. arXiv preprint arXiv:190204166, https://doi.org/10.1093/bioinformatics/bty593
7. Wang YB, Yi HC, Yang S et al (2020) A deep learning-based method for drug-target interaction prediction based on long short-term memory neural network. BMC Med Inform Decis Mak 20:49. https://doi.org/10.1186/s12911-020-1052-0
8. Karimi M, Wu D, Wang Z et al (2019) DeepAffinity: interpretable deep learning of compound-protein affinity through unified recurrent and convolutional neural networks. Bioinformatics 35:3329–3338. https://doi.org/10.1093/bioinformatics/btz111
9. Nguyen T, Le H, Quinn TP et al (2021) GraphDTA: predicting drug–target binding affinity with graph neural networks. Bioinformatics 37:1140–1147. https://doi.org/10.1093/bioinformatics/btaa921
10. Jiang M, Li Z, Zhang S et al (2020) Drug–target affinity prediction using graph neural network and contact maps. RSC Adv 10:20701–20712. https://doi.org/10.1039/D0RA02297G
11. Zhang S, Jiang M, Wang S et al (2021) SAG-DTA: prediction of drug-target affinity using self-attention graph network. Int J Mol Sci 22:8993. https://doi.org/10.3390/ijms22168993
12. Yang Z, Zhong W, Zhao L, Chen CY-C (2022) MGraphDTA: deep multiscale graph neural network for explainable drug–target

binding affinity prediction. Chem Sci 13:816–833. https://doi.org/10.1039/D1SC05180F
13. Zeng X, Zhong KY, Meng PY et al (2024) MvGraphDTA: multi-view-based graph deep model for drug-target affinity prediction by introducing the graphs and line graphs. BMC Biol 22:1–13. https://doi.org/10.1186/s12915-024-01981-3
14. Li Y, Li P, Sun D, Liu ZP PGDTA: predicting drug-target affinity using three-dimensional structure of protein pocket and graph neural network. IEEE Trans Comput Biol Bioinform 2998–4165. https://doi.org/10.1109/TCBBIO.2025.3563504
15. Chu Z, Liu S, Zhang W (2022) Hierarchical graph representation learning for the prediction of drug-target binding affinity. Inf Sci 613:507–523. https://doi.org/10.1016/j.ins.2022.09.043
16. Ma W, Zhang S, Li Z et al (2023) Predicting drug-target affinity by learning protein knowledge from biological networks. IEEE J Biomed Health Inform 27:2128–2137. https://doi.org/10.1109/JBHI.2023.3240305
17. Yunan L, Yang L, Jian P (2023) Calibrated geometric deep learning improves kinase–drug binding predictions. Nat Mach Intell 5:1390–1401. https://doi.org/10.1038/s42256-023-00751-0
18. Wang S, Zhang Y, Liang D et al (2025) TarMGDif: target-specific molecular graphs generation based on diffusion model. IEEE J Biomed Health Inform 15:35872. https://doi.org/10.1109/JBHI.2025.3569105
19. Liang D, Yu R, Wang X et al (2025) MTMP: Multimodal targeted molecule generation model with protein features. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2025.129845
20. Zhang Z, Li Z, Li W et al (2025) EDG-PPIS: an equivariant and dual-scale graph network for protein–protein interaction site prediction. BMC Genomics 26:862. https://doi.org/10.1186/s12864-025-12084-w
21. Khan S, Dilshad N, Ahmad N et al (2025) Integrating AI in security information and event management for real time cyber defense. Sci Rep 15:35872. https://doi.org/10.1038/s41598-025-19689-x
22. Noor S, AlQahtani SA, Khan S (2025) XGBoost-Liver: an intelligent integrated features approach for classifying liver diseases using ensemble XGBoost training model. Comput Mater Contin. https://doi.org/10.32604/cmc.2025.061700
23. Khan S, Noor S, Awan HH et al (2025) Deep-ProBind: binding protein prediction with transformer-based deep learning model. BMC Bioinformatics 26:88. https://doi.org/10.1186/s12859-025-06101-8
24. Almusallam N, Khan S, Alarfaj FK, Ahmad N (2025) A robust deep learning framework for RNA 5-methyluridine modification prediction using integrated features. BMC Biol 23:1–15. https://doi.org/10.1186/s12915-025-02433-2
25. Wang L, Wong L, Chen Z-H et al (2022) MSPEDTI: prediction of drug–target interactions via molecular structure with protein evolutionary information. Biology 11:740. https://doi.org/10.3390/biology11050740
26. D'Souza S, Prema KV, Balaji S, Shah R (2023) Deep learning-based modeling of drug–target interaction prediction incorporating binding site information of proteins. Interdiscip Sci 15:306–315. https://doi.org/10.1007/s12539-023-00557-z
27. Voitsitskyi T, Stratiichuk R, Koleiev I et al (2023) 3DProtDTA: a deep learning model for drug-target affinity prediction based on residue-level protein graphs. RSC Adv 13:10261–10272. https://doi.org/10.1039/D3RA00281K
28. Klipp E, Wade RC, Kummer U (2010) Biochemical network-based drug-target prediction. Curr Opin Biotechnol 21:511–516. https://doi.org/10.1016/j.copbio.2010.05.004
29. Rifaioglu AS, Cetin Atalay R, Cansen Kahraman D et al (2021) MDeePred: novel multi-channel protein featurization for deep learning-based binding affinity prediction in drug discovery. Bioinformatics 37:693–704. https://doi.org/10.1093/bioinformatics/btaa858
30. Zhang L, Wang C-C, Zhang Y, Chen X (2023) GPCNDTA: prediction of drug-target binding affinity through cross-attention networks augmented with graph features and pharmacophores. Comput Biol Med 166:107512. https://doi.org/10.1016/j.compbiomed.2023.107512
31. Li J, Bi X, Ma W et al (2024) MHAN-DTA: a multiscale hybrid attention network for drug-target affinity prediction. IEEE J Biomed Health Inform. https://doi.org/10.1109/JBHI.2024.3518619
32. Pang X, Liu Z, Chen Q et al (2025) A multimodal DTA prediction method based on triple-view contrastive learning. Complex Intell Syst 11:387. https://doi.org/10.1007/s40747-025-02020-6
33. Gao Z, Jiang C, Zhang J et al (2023) Hierarchical graph learning for protein–protein interaction. Nat Commun 14:1093. https://doi.org/10.1038/s41467-023-36736-1
34. Szklarczyk D, Gable AL, Nastou KC et al (2021) The STRING database in 2021: customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets. Nucleic Acids Res 49:D605–D612. https://doi.org/10.1093/nar/gkaa1074
35. Landrum G (2013) RDKit: a software suite for cheminformatics, computational chemistry, and predictive modeling. Greg Landrum 8:5281
36. Ma W, Zhang S, Li Z et al (2022) Enhancing protein function prediction performance by utilizing AlphaFold-predicted protein structures. J Chem Inf Model 62:4008–4017. https://doi.org/10.1021/acs.jcim.2c00885
37. Grover A, Leskovec J (2016) node2vec: Scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, pp 855–864. https://doi.org/10.1145/2939672.2939754
38. Gou J, Yu B, Maybank SJ, Tao D (2021) Knowledge distillation: a survey. Int J Comput Vis 129:1789–1819. https://doi.org/10.1007/s11263-021-01453-z
39. Davis MI, Hunt JP, Herrgard S et al (2011) Comprehensive analysis of kinase inhibitor selectivity. Nat Biotechnol 29:1046–1051. https://doi.org/10.1038/nbt.1990
40. Tang J, Szwajda A, Shakyawar S et al (2014) Making sense of large-scale kinase inhibitor bioactivity data sets: a comparative and integrative analysis. J Chem Inf Model 54:735–743. https://doi.org/10.1021/ci400709d
41. Liu H, Sun J, Guan J et al (2015) Improving compound–protein interaction prediction by building up highly credible negative samples. Bioinformatics 31:i221–i229. https://doi.org/10.1093/bioinformatics/btv256
42. Bemis GW, Murcko MA (1996) The properties of known drugs. 1. Molecular frameworks. J Med Chem 39:2887–2893. https://doi.org/10.1021/jm9602928
43. Lv Q, Zhou J, Yang Z et al (2023) 3D graph neural network with few-shot learning for predicting drug–drug interactions in scaffold-based cold start scenario. Neural Netw 165:94–105. https://doi.org/10.1016/j.neunet.2023.05.039
44. Lv Q, Chen G, Yang Z et al (2023) Meta learning with graph attention networks for low-data drug discovery. IEEE Trans Neural Netw Learn Syst 35:11218–11230. https://doi.org/10.1109/TNNLS.2023.3250324
45. Lv Q, Chen G, Yang Z et al (2024) Meta-molnet: a cross-domain benchmark for few examples drug discovery. IEEE Trans Neural Netw Learn Syst 36:4849–4863. https://doi.org/10.1109/TNNLS.2024.3359657
46. Perozzi B, Al-Rfou R, Skiena S (2014) Deepwalk: Online learning of social representations. In: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and

data mining, pp 701–710. https://doi.org/10.1145/2623330.2623732
47. Tang J, Qu M, Wang M, et al (2015) Line: Large-scale information network embedding. In: Proceedings of the 24th international conference on world wide web, pp 1067–1077. https://doi.org/10.1145/2736277.2741093
48. Öztürk H, Ozkirimli E, Özgür A (2019) WideDTA: prediction of drug-target binding affinity. arXiv preprint arXiv:190204166 https://doi.org/10.1093/bioinformatics/bty593
49. Metz JT, Johnson EF, Soni NB et al (2011) Navigating the kinome. Nat Chem Biol 7:200–202. https://doi.org/10.1038/nchembio.530
50. Chen L, Tan X, Wang D et al (2020) Transformercpi: improving compound–protein interaction prediction by sequence-based deep learning with self-attention mechanism and label reversal experiments. Bioinformatics 36:4406–4414. https://doi.org/10.1093/bioinformatics/btaa524
51. Chen G, He H, Zhao L et al (2024) GINCM-DTA: a graph isomorphic network with protein contact map representation for potential use against COVID-19 and Omicron subvariants BQ. 1, BQ. 1.1, XBB. 1.5, XBB. 1.16. Expert Syst Appl 236:121274. https://doi.org/10.1016/j.eswa.2023.121274
52. Lawler GF, Limic V (2010) Random walk: a modern introduction. Cambridge University Press. https://doi.org/10.1017/CBO9780511750854
53. Hamilton W, Ying Z, Leskovec J (2017) Inductive representation learning on large graphs. Adv Neural Inf Process Syst 30
54. Chiang W-L, Liu X, Si S, et al (2019) Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, pp 257–266. https://doi.org/10.1145/3292500.3330925
55. Liao J, Chen H, Wei L, Wei L (2022) Gsaml-dta: an interpretable drug-target binding affinity prediction model based on graph neural networks with self-attention mechanism and mutual information. Comput Biol Med 150:106145. https://doi.org/10.1016/j.compbiomed.2022.106145
56. von Mering C, Huynen M, Jaeggi D et al (2003) STRING: a database of predicted functional associations between proteins. Nucleic Acids Res 31:258–261. https://doi.org/10.1093/nar/gkg034
57. Alturki R, Munshi A, Alshawi B et al (2025) CardioBERT: a cardiac identification using fusion features in consumer healthcare. IEEE Trans Consum Electron. https://doi.org/10.1109/TCE.2025.3575522
58. Almusallam N, Khan S (2025) Chronic liver disease classification using deep learning with SHAP-optimized hybrid features. iScience. https://doi.org/10.1016/j.isci.2025.113972
59. Noor S, Awan HH, Hashmi AS et al (2025) Optimizing performance of parallel computing platforms for large-scale genome data analysis. Computing 107:1–22. https://doi.org/10.1007/s00607-025-01441-y