



DynamicDTA: Drug-Target Binding Affinity Prediction Using Dynamic Descriptors and Graph Representation

Dan Luo¹ · Jinyu Zhou¹ · Le Xu¹ · Sisi Yuan² · Xuan Lin^{1,3}

Received: 12 February 2025 / Revised: 7 May 2025 / Accepted: 11 May 2025
 © International Association of Scientists in the Interdisciplinary Areas 2025

Abstract

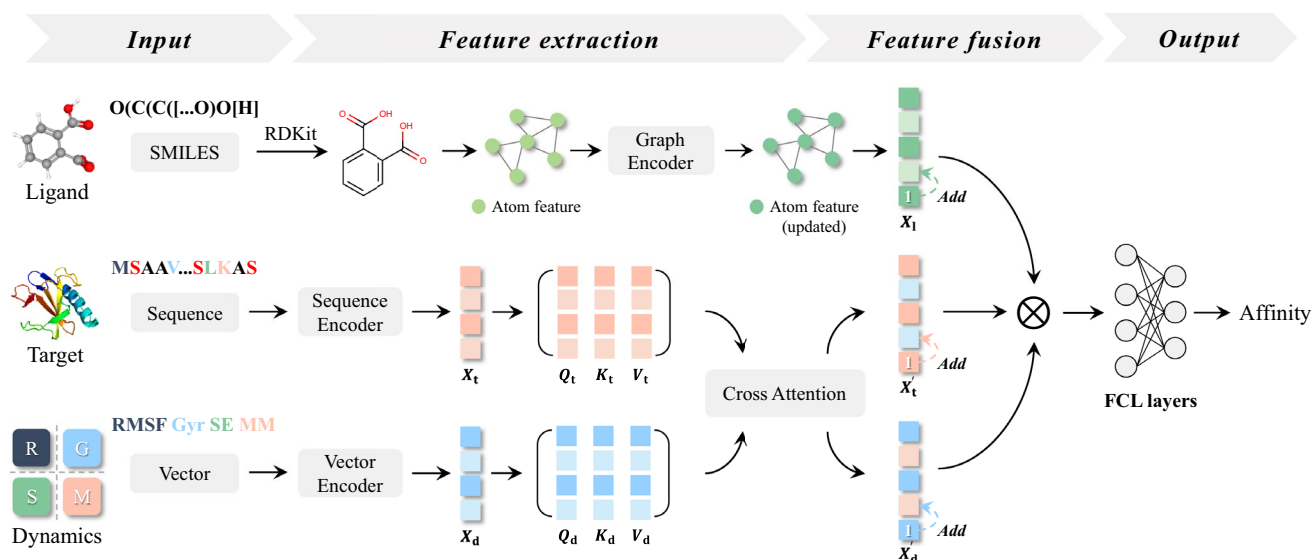
Motivation Predicting drug–target binding affinity (DTA) is essential for identifying potential therapeutic candidates in drug discovery. However, most existing models rely heavily on static protein structures, often overlooking the dynamic nature of proteins, which is crucial for capturing conformational flexibility that will be beneficial for protein binding interactions.

Methods We introduce DynamicDTA, an innovative deep learning framework that incorporates static and dynamic protein features to enhance DTA prediction. The proposed DynamicDTA takes three types of inputs, including drug sequence, protein sequence, and dynamic descriptors. A molecular graph representation of the drug sequence is generated and subsequently processed through graph convolutional network, while the protein sequence is encoded using dilated convolutions. Dynamic descriptors, such as root mean square fluctuation, are processed through a multi-layer perceptron. These embedding features are fused with static protein features using cross-attention, and a tensor fusion network integrates all three modalities for DTA prediction.

Results Extensive experiments on three datasets demonstrate that DynamicDTA achieves by at least 3.4% improvement in e_{RMSE} score with comparison to seven state-of-the-art baseline methods. Additionally, predicting novel drugs for *Human Immunodeficiency Virus Type 1* and visualizing the docking complexes further demonstrates the reliability and biological relevance of DynamicDTA.

Availability and implementation The source code is publicly available and can be accessed at <https://github.com/shmily-ld/DynamicDTA>.

Graphical Abstract



Extended author information available on the last page of the article

Keywords Drug discovery · Drug-target binding affinity · Protein dynamics · Deep learning

1 Introduction

An initial step in the drug discovery pipeline [1] is to identify molecules that bind to specific protein targets with high affinity and specificity, which can be further developed into drug-like molecules [2]. Drug-target binding affinity (DTA) is a crucial metric that quantifies the strength of interaction between a pair of drug and target protein, playing a pivotal role in the efficacy and specificity of potential therapeutic compounds [3]. Accurate prediction of DTA is essential in drug discovery [4, 5], as it facilitates virtual screening, lead optimization, and toxicity mitigation [6]. By prioritizing compounds with high affinity, guiding structural modifications, and predicting off-target interactions, DTA prediction reduces biochemical validation costs and accelerates the identification of druggable candidates, ultimately improving the efficiency of the drug discovery [7, 8].

Traditional experimental methods for measuring binding affinity, such as isothermal titration calorimetry and surface plasmon resonance, provide precise measurements but are often time-consuming, resource-intensive, and impractical for large-scale screening. To overcome these limitations, computational methods have emerged as efficient alternatives [9, 10]. These methods leverage data from public repositories like Kiba [11] and Davis [12] to predict binding affinities by leveraging the molecular and structural characteristics of drugs and proteins. Among these computational strategies, deep learning has become a powerful approach [13], owing to its unparalleled capacity to model complex, non-linear relationships between input features [14].

Within the domain of deep learning-based DTA prediction, methods can be broadly categorized into sequence-based and structure-based approaches. Sequence-based models utilize the primary sequence information of drugs and proteins, providing a high-level abstraction of molecular characteristics. A seminal work in this domain, DeepDTA [15], demonstrated the utility of convolutional neural network (CNN) in capturing local sequence patterns and interactions, achieving substantial improvements over traditional methods. WideDTA [16] is further proposed to integrate four sources of text-based information. AttentionDTA [17] combines an attention mechanism with the prediction of binding affinity, enhancing the extraction of relevant features from both drugs and protein sequences, resulting in a comprehensive deep learning-driven framework. Unlike static word embedding, DEAttentionDTA [18] integrates dynamic word encoding with multi-head self-attention, enabling multi-scale feature interaction between drugs and target proteins.

In contrast, structure-based models capitalize on the three-dimensional conformational information of drugs and proteins to provide a more granular representation of molecular interactions [19]. Such as, GraphDTA [20] employs graph neural network (GNN) to model molecular graphs, thereby facilitating a more nuanced understanding of molecular interactions. Similarly, DGraphDTA [21] represents proteins as contact maps and utilizes GNN to simultaneously learn the features of both drugs and proteins, enhancing the accuracy of interaction predictions. ImageDTA [22] focuses on the 2D structure of drugs, treating drug 2D representations as “images” and processing them with multiscale 2D-CNNs for better interpretability and performance. The known protein sequences [23] number in the billions, while the currently identified protein structures [24] represent only a small portion. In practice, obtaining the accurate 3D structure of the drug–target complex or even the protein itself is often challenging [25, 26]. Consequently, our work exclusively employs molecular graphs, bypassing the use of protein graphs.

A common limitation of most existing approaches is their reliance on static features of proteins, which fail to capture the inherently dynamic nature of these biomolecules. Proteins are not rigid entities, and they undergo continuous structural fluctuations, including changes in atomic positions and binding postures, driven by their biological environment and functional role. These dynamic characteristics are critical for understanding key processes such as molecular recognition, allosteric regulation, and binding interactions [27]. By affecting the accessibility, shape, and compatibility of binding sites, protein dynamics directly influence the specificity and strength of drug–target interactions [28]. Extracting dynamic features allows models to capture this vital information, leading to more accurate predictions and a deeper understanding of protein function, ultimately facilitating more effective drug discovery.

To address this challenge, we propose DynamicDTA, an innovative framework designed to predict DTA by incorporating dynamic protein features derived from molecular dynamics (MD) simulations [29]. By integrating time-dependent structural descriptors, such as root mean square fluctuation (RMSF) [30], DynamicDTA provides a more realistic representation of protein behavior. These dynamic features allow the model to capture temporal variations in protein conformations, thereby enhancing its ability to predict binding affinity. DynamicDTA introduces several innovations to advance the state-of-the-art methods in DTA prediction. First, it incorporates dynamic features derived

from molecular dynamics simulations into the model input. Second, it uses a cross-attention mechanism to learn both static and dynamic protein features, enabling the model to capture their interactions more effectively. Additionally, a tensor fusion network (TFN) [31] is employed to integrate multi-modal information from both drugs and proteins, ensuring comprehensive feature fusion. Comprehensive experiments have shown that DynamicDTA consistently outperforms seven state-of-the-art methods. Overall, our contributions are summarized as follows:

- We introduce DynamicDTA, an innovative framework that incorporates dynamic protein features derived from MD simulations, offering a more precise representation of protein behavior for predicting binding affinity.
- We leverage a cross-attention mechanism to capture both static and dynamic protein features, while a TFN seamlessly integrates multi-modal data from both drugs and proteins, enabling the model to effectively combine diverse feature representations and enhance feature fusion.
- Comparison experiments on three datasets have shown that DynamicDTA outperforms seven state-of-the-art methods. Furthermore, a case study predicting potential drugs for *Human Immunodeficiency Virus Type 1* showcases the model's potential in accelerating drug discovery.

2 Materials and Methods

2.1 Datasets

The protein dynamic descriptors are sourced from ATLAS dataset [32], which currently includes 1938 proteins and provides four types of protein dynamic features from MD simulations:

- (1) Avg.RMSF (average RMSF): This descriptor measures the fluctuation of each atom or residue from its average position during MD simulations. Specifically, it quantifies the root mean square deviation of atomic positions over time, providing insights into the flexibility of individual residues or regions within a protein. Typically, residues with high Avg.RMSF values are found in flexible regions such as loops or unstructured coils, which may be involved in essential protein functions like substrate binding, enzymatic regulation, or protein-protein interactions [33]. Conversely, residues with low Avg.RMSF values are often located in rigid regions such as α -helices and β -sheets, which maintain structural stability and play a crucial role in preserving the overall protein conformation [34]. In drug design,

regions with high RMSF values could serve as potential drug targets due to their dynamic adaptability in drug–target interactions [35].

- (2) Avg.Gyr (average gyration radius): Gyr measures the overall compactness of the protein structure by calculating the average distance between the protein's atoms and its center of mass. Avg.Gyr over the simulation period provides insight into the stability and folding of the protein. A smaller Avg.Gyr suggests a more compact structure, which may indicate an active conformation or a stable binding state with other molecules. Conversely, a larger Avg.Gyr may suggest a more extended protein conformation, potentially associated with dynamic processes such as folding, unfolding, or dissociation from other molecules [36]. Changes in the radius of gyration can reflect dynamic interactions and conformational transitions within the structure, which are essential for understanding their functional mechanisms.
- (3) Div.SE (minimum TM-score between start and final conformations): This descriptor quantitatively evaluates the structural divergence between the initial and final conformations of a protein by employing TM-score [37], which is a widely used metric for assessing the similarity of protein structures. A high Div.SE value indicates minimal structural changes during the simulation, suggesting that the protein maintains a stable conformation, which may imply high structural rigidity under physiological conditions, allowing it to perform its biological function effectively. Conversely, a low Div.SE value suggests significant conformational changes, which could be associated with the protein's dynamic functionality, such as structural adaptations required for substrate binding or catalytic processes in enzymatic reactions [38, 39].
- (4) Div.MM (minimum TM-score between most divergent conformations): This descriptor assesses the structural diversity of a protein by measuring TM-score between the most divergent conformations observed during the simulation. It captures the protein's flexibility by identifying how much its structure changes between its most extreme conformations. A high Div.MM value suggests that, while the protein undergoes conformational changes, its overall structure remains similar, indicating flexibility without drastic rearrangements. Conversely, a low Div.MM value suggests significant conformational shifts, potentially linked to protein folding, allosteric regulation, or molecular interactions [40].

The affinity data are obtained from BindingDB [41], a comprehensive database of experimentally measured binding affinities, which contains 2.9 million entries across 1.3 million compounds and 9400 targets. However, some

data points for K_i , K_d , and IC_{50} , which are commonly used metrics to measure binding affinity [42], were missing in the dataset. To address this issue, we split the data into three subsets based on different affinity measures. Additionally, to ensure that the affinity values fall within a suitable range for modeling, we applied a negative logarithmic transformation. Taking K_d as an example, the transformation is applied using the following formula:

$$K'_d = -\lg\left(\frac{K_d}{1 \times 10^9}\right) \quad (1)$$

After the negative logarithmic transformation, some of the resulting values were negative. These values were removed, as they do not align with the expected range of affinity values. Finally, the values for K'_d range from 0.3 to 11.9, K'_i from 0.0 to 14.2, and IC'_{50} from 0.0 to 12.6.

To integrate the protein dynamic features with the affinity data, we perform a matching process based on PDB ID. As a result, three final datasets were created, named as K_d^* , K_i^* , and IC_{50}^* . Table 1 displays a summary of the key details of the three datasets.

Table 1 Summary of the datasets

| Dataset | Targets | Ligands | Binding entities |
|-------------|---------|---------|------------------|
| K_d^* | 63 | 1609 | 2896 |
| K_i^* | 74 | 18,872 | 27,106 |
| IC_{50}^* | 136 | 45,525 | 86,236 |
| Kiba* | 4 | 1597 | 3708 |

We evaluated the model using five-fold cross-validation, as in AttentionDTA [17]. Each dataset was split into five parts, using one for testing and the rest for training in each fold. The final result were averaged across all folds. To further assess the model's generalization ability, we additionally processed Kiba dataset [11] following the same procedure and ensuring that no proteins or drugs overlap with those in other datasets, resulting in a new dataset named Kiba*. This external dataset was used to confirm the model's ability to generalize to unseen data, providing further validation of its robustness.

2.2 DynamicDTA Framework

As depicted in Fig. 1, the DynamicDTA framework consists of four key components: input representation, feature extraction, feature fusion, and output block. In the following sections, we will describe each part of the framework in detail.

2.2.1 Input Representation

The DynamicDTA model takes three types of input data: ligands, targets, and dynamics. In what follows, we provide a detailed description of each input representation.

Ligand representation. Ligands in this study are represented using the simplified molecular input line entry system (SMILES), a standardized and compact notation for encoding molecular structures as linear text strings [43]. To utilize the structural and chemical information captured by SMILES, we convert these strings into graph-based representations. The process begins by parsing each SMILES string into a molecular graph using the open-source chemical

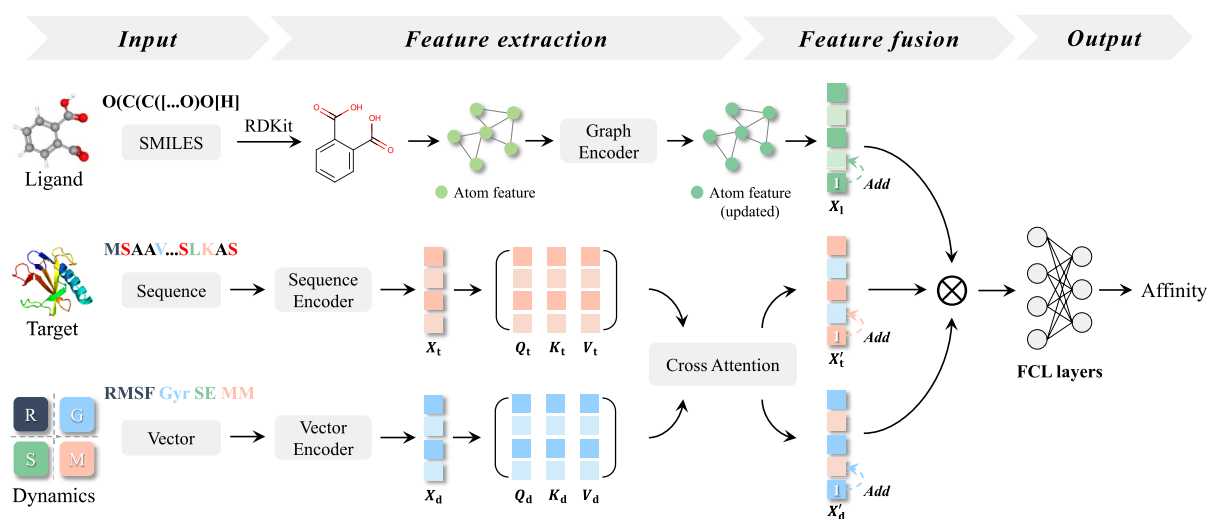


Fig. 1 The architecture of DynamicDTA. The framework integrates drug molecular graph, target sequences and dynamic descriptors, and a cross-attention mechanism to extract meaningful representations.

TFN is employed to effectively fuse the extracted features for accurate drug–target binding affinity prediction

informatics software RDKit [44]. In this graph, atoms serve as nodes, while chemical bonds act as edges. Each atom is represented by a feature vector that encapsulates critical atomic properties, including atomic number, valence, and aromaticity. These feature vectors are normalized to maintain numerical stability across different molecular graphs. Bond information is extracted as pairs of indices indicating whether there is a bond between two atoms in the graph.

Target representation. For target protein representation, each protein sequence is converted into a numerical vector using a predefined mapping of amino acid characters to integers. Each amino acid residue is assigned a unique integer, and characters not found in the mapping are assigned a default value of 0. To address the variation in protein sequence lengths, we define a fixed maximum length of 1000. Sequences longer than this threshold are truncated, while shorter sequences are padded with 0. This standardization ensures that all protein sequences have the same length, enabling efficient batch processing and maintaining consistent input dimensions for the model.

Dynamic representation. In this study, we focus on four key dynamic features to represent the dynamic nature of proteins: Avg.RMSF, Avg.Gyr, Div.SE, and Div.MM. To standardize these dynamic descriptors and make them comparable across proteins, we perform min-max normalization on each feature. Specifically, for a feature x , the normalization is defined as

$$x_{\text{norm}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (2)$$

where x_{\min} and x_{\max} represent the minimum and maximum values of feature x across the dataset. After normalization, the four dynamic features are concatenated into a single 4-dimensional vector. This vector serves as a comprehensive representation of the protein's dynamic behavior, capturing key aspects such as flexibility, structural transitions, and potential binding site changes, which are crucial for understanding drug–target interactions. The integration of these dynamic features improves the model's capacity to predict binding affinities by accounting for the protein's conformational flexibility and structural transitions, making this approach a significant innovation in DTA prediction.

2.2.2 Feature Extraction

Graph Encoder. To extract meaningful features from the graph representation of a ligand, we utilize a graph convolutional network (GCN) to learn a graph-level representation that represents its molecular architecture. Given a ligand represented by graph $G = (V, E)$, where V is the set of N nodes, each node has an associated feature vector $\mathbf{x}_i \in \mathbb{R}^C$, and E is the set of edges described by the adjacency matrix

$\mathbf{A} \in \mathbb{R}^{N \times N}$. GCN uses the node feature matrix \mathbf{X} and the adjacency matrix \mathbf{A} as inputs. The propagation rule for GCN is given by

$$\mathbf{H}^{(l+1)} = \sigma(\tilde{\mathbf{D}}^{-1/2} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-1/2} \mathbf{H}^{(l)} \mathbf{W}^{(l)}) \quad (3)$$

where $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_N$ is the adjacency matrix with self-connections, $\tilde{\mathbf{D}}$ is the degree matrix of $\tilde{\mathbf{A}}$, $\mathbf{W}^{(l)}$ is the learnable weight matrix at layer l , and σ is a non-linear activation function. After several layers, a global max pooling operation is applied to obtain a final feature vector of ligand \mathbf{X}_l .

Sequence Encoder. Previous studies have shown that dilated convolution [45] is an effective technique for capturing multiscale contextual information by expanding the receptive field through different dilation rates [46]. Inspired by this, we apply dilated 1D convolutions to model long-range intramolecular interactions in protein sequences. While a standard 1D convolution operates on adjacent elements of the sequence with a fixed receptive field, the dilated 1D convolution increases the spacing between the filter elements, allowing it to effectively capture dependencies over long distances in the sequence without increasing the number of parameters. The dilated convolution operation for a given input sequence $S = \{s_1, s_2, \dots, s_N\}$, where s_i represents the feature vector of amino acid i , is performed as follows:

$$\mathbf{h}_i^{(l)} = \sigma\left(\sum_{j=1}^K \mathbf{w}_j^{(l)} s_{i+r_j} + \mathbf{b}^{(l)}\right) \quad (4)$$

where $\mathbf{w}_j^{(l)}$ are the weights of the convolutional filter at layer l , r_j denotes the dilation rate for the j -th filter, K is the filter size, and $\mathbf{b}^{(l)}$ is the bias term. Similar to the ligand representation, after passing through the dilated convolutions, the final feature vector of target \mathbf{X}_t is obtained by applying a global max pooling operation.

Vector Encoder. The dynamic features of each protein are represented as a normalized 4-dimensional vector \mathbf{V}_d . To extract high-level representations, the normalized vector \mathbf{V}_d is passed through a multi-layer perceptron (MLP). The transformation at each MLP layer is defined as

$$\mathbf{X}_d^{(l+1)} = \sigma(\mathbf{W}^{(l)} \mathbf{X}_d^{(l)} + \mathbf{b}^{(l)}) \quad (5)$$

where $\mathbf{X}_d^{(0)} = \mathbf{V}_d$ is the input vector, $\mathbf{W}^{(l)}$ and $\mathbf{b}^{(l)}$ represent the weight matrix and bias vector for the l -th layer, respectively. The final output \mathbf{X}_d is a compact feature vector that captures the essential information from the dynamic descriptors.

Cross Attention Mechanism. To effectively integrate complementary information between the target sequence vector \mathbf{X}_t and the dynamic vector \mathbf{X}_d , we employ a multi-head cross attention mechanism, which captures bidirectional interactions, enabling \mathbf{X}_t to utilize information from

X_d , and X_t to incorporate information from X_l . Initially, the target and dynamic vectors are linearly transformed to compute their respective query, key, and value matrices for each head:

$$Q_i^{(i)} = W_Q^{t(i)} X_t \quad (6)$$

$$K_d^{(i)} = W_K^{d(i)} X_d \quad (7)$$

$$V_d^{(i)} = W_V^{d(i)} X_d \quad (8)$$

where each head $i = 1, \dots, H$, where H is the number of attention heads, and $W_Q^{t(i)}$, $W_K^{d(i)}$, and $W_V^{d(i)}$ are learnable weight matrices. The attention computation for the i -th head is given by

$$A_t^{(i)} = \text{Softmax} \left(\frac{Q_i^{(i)} (K_d^{(i)})^\top}{\sqrt{d_K}} \right) V_d^{(i)} \quad (9)$$

where d_K is the dimension of the key matrices. The multi-head outputs are concatenated along the feature dimension and then passed through a learned linear transformation:

$$X_t' = \text{Concat}(A_t^{(1)}, \dots, A_t^{(H)}) \cdot W_t^O \quad (10)$$

where W_t^O is the output projection matrix. Similarly, to enable the dynamic vector X_d to attend to the target vector X_t , the corresponding query, key, and value matrices for each head are computed as

$$Q_d^{(i)} = W_Q^{d(i)} X_d \quad (11)$$

$$K_t^{(i)} = W_K^{t(i)} X_t \quad (12)$$

$$V_t^{(i)} = W_V^{t(i)} X_t \quad (13)$$

The attention computation for the i -th head is

$$A_d^{(i)} = \text{Softmax} \left(\frac{Q_d^{(i)} (K_t^{(i)})^\top}{\sqrt{d_K}} \right) V_t^{(i)}. \quad (14)$$

The multi-head outputs are concatenated along the feature dimension and then passed through a learned linear transformation:

$$X_d' = \text{Concat}(A_d^{(1)}, \dots, A_d^{(H)}) \cdot W_d^O \quad (15)$$

The model effectively fuses complementary information from both vectors using the multi-head cross-attention mechanism, attending to each other in parallel and enriching their representations with contextual dependencies. This

results in more informative and context-aware feature representations for both the target sequence and dynamic features.

2.2.3 Feature Fusion

We use TFN [31] to integrate the features X_t' , X_d' , and X_l from the target, dynamics, and ligand, respectively. Specifically, we augment each modality's representation by adding an extra constant dimension with a value of 1. The extended representations are given by

$$\tilde{X}_l = \begin{bmatrix} X_l \\ 1 \end{bmatrix} \quad \tilde{X}_t = \begin{bmatrix} X_t' \\ 1 \end{bmatrix} \quad \tilde{X}_d = \begin{bmatrix} X_d' \\ 1 \end{bmatrix}. \quad (16)$$

Next, we fuse the information from the three extended feature vectors, \tilde{X}_l , \tilde{X}_t , and \tilde{X}_d , by computing their outer product:

$$X_f = W_f \cdot (\tilde{X}_t \otimes \tilde{X}_d \otimes \tilde{X}_l) + b_f \quad (17)$$

where \otimes denotes the outer product operation, $\tilde{X}_t \otimes \tilde{X}_d \otimes \tilde{X}_l$ is the higher-order tensor that captures interactions across unimodal, bimodal, and trimodal features, and W_f and b_f are learnable parameters. X_f represents the final fused feature. TFN explicitly captures the complex interactions between ligand, target and dynamic features, facilitating the seamless integration of multimodal information into a comprehensive, high-dimensional feature representation.

2.2.4 Prediction and Training Module

The prediction module employs fully connected layers (FCLs) to process the fused feature vector X_f obtained from TFN. ReLU is used as the activation function in each layer to capture complex nonlinear patterns, leading to the prediction of the final binding affinity score \hat{y} . DTA prediction is framed as a regression problem [47]. The model is optimized to minimize the mean squared error (MSE) loss function, which is defined as

$$\mathcal{L} = \frac{1}{M} \sum_{i=1}^M (y_i - \hat{y}_i)^2 \quad (18)$$

where y_i denotes the actual binding affinity of the i -th sample, \hat{y}_i represents the predicted binding affinity, while M represents the total number of samples.

3 Result

In this section, we introduce and evaluate the performance of model with comparison to baseline methods, followed by a case study that demonstrates its practical application.

3.1 Experimental Settings

The DynamicDTA model was implemented using PyTorch and trained on a Linux server with 12 CPUs and an Nvidia GeForce RTX 4090 GPU with 24 GB of VRAM. We set the number of epochs to 1000, the batch size to 512, and the learning rate to 5×10^{-4} , respectively. These hyperparameters were chosen to ensure efficient training and convergence of the model. Additionally, we set the number of GCN hidden layers to 3, the attention heads to 4, the embedding dimension to 64 and the dropout rate to 0.2 to prevent overfitting, respectively. Other detailed model parameters can be found in the source code. The Adam optimizer [48] was used to update the model parameters, and MSE was employed as the loss function. To ensure robust evaluation, five-fold cross-validation was performed across all datasets.

3.2 Baseline Methods

To assess the performance of our model, we compared the proposed DynamicDTA with seven baseline methods for DTA prediction, including two traditional machine learning models implemented with scikit-learn library and five state-of-the-art deep learning-based approaches. The deep learning baselines were implemented using the source code provided in the original papers, while the traditional machine learning models were built using standard implementations, without any additional tuning. The seven models used for comparison are the following:

- **Linear Regression** employs label encoding on drug SMILES and protein sequences, using concatenated vectors for linear affinity prediction.
- **Decision Tree** applies the same label encoding followed by decision tree regression with feature space partitioning.
- **DeepDTA** [15] employs CNN to process protein sequences and drug SMILES representations for DTA prediction.
- **GraphDTA** [20] employs GNN to extract and learn molecular graph structural representations, combined with protein sequence embeddings for binding affinity prediction.
- **AttentionDTA** [17] employs an attention-based architecture to enhance the extraction of relevant features from drug SMILES and protein sequences.
- **DEAttentionDTA** [18] uses a 1D CNN for dynamic word embedding and captures interactions between drugs and proteins features by integrating a self-attention mechanism.

- **ImageDTA** [22] leverages a multiscale 2D CNN to process SMILES-encoded molecules and enhances interpretability through convolutional kernel size selection.

3.3 Evaluation Metrics

To quantitatively evaluate the predictive performance of our DTA model, we employ two widely used metrics, the root mean square error (RMSE, denoted by e_{RMSE}) as well as the Pearson correlation coefficient (R). The specific formulas are as follows:

$$e_{\text{RMSE}} = \sqrt{\frac{1}{M} \sum_{i=1}^M (y_i - \hat{y}_i)^2} \quad (19)$$

$$R = \frac{\sum_{i=1}^M (y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum_{i=1}^M (y_i - \bar{y})^2 \sum_{i=1}^M (\hat{y}_i - \bar{\hat{y}})^2}} \quad (20)$$

where e_{RMSE} measures the average magnitude of prediction errors, with lower values indicating better prediction accuracy. R measures the linear relationship between the predicted and true values, with higher values indicating a stronger correlations between them.

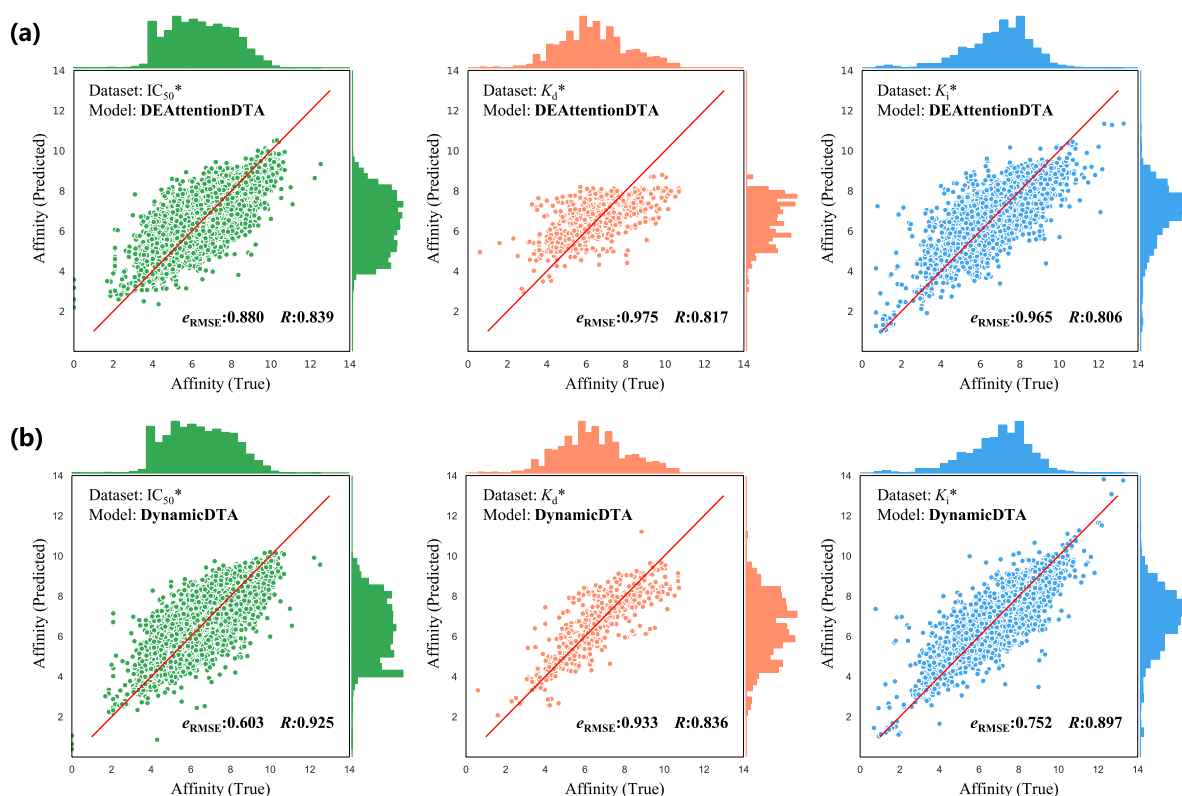
3.4 Comparison Results

Table 2 summarizes the comparison performance of DynamicDTA with seven baseline methods on three datasets. The results demonstrate that our model consistently outperforms all baseline methods. On the K_d^* dataset, DynamicDTA achieves a notable improvement, surpassing GraphDTA, the second-best model, by 2.2% in e_{RMSE} and 1.4% in R . Similarly, on the K_i^* dataset, DynamicDTA again outperforms the second-best method, with improvement of 6.5% in e_{RMSE} and 2.6% in R . However, on the IC_{50}^* dataset, while DynamicDTA achieves the highest R , it falls slightly behind ImageDTA in e_{RMSE} . This discrepancy may stem from the nature of the IC_{50} metric, which focuses more on a drug's inhibitory effects rather than directly correlating with binding affinity [49, 50]. Furthermore, ImageDTA places greater emphasis on drug-specific features, which could explain its slight advantage in this specific metric. Traditional machine learning models perform poorly compared to deep learning-based methods like DeepDTA due to their reliance on hand-crafted features, which may not fully capture the complex interactions between drugs and proteins. In contrast, deep learning models automatically extract hierarchical features,

Table 2 Comparison performance of DynamicDTA and baseline methods on three datasets

| Model | IC_{50}^* | | K_d^* | | K_i^* | |
|-------------------|----------------------------|---------------------|----------------------------|---------------------|----------------------------|---------------------|
| | $e_{RMSE}(std) \downarrow$ | $R(std) \uparrow$ | $e_{RMSE}(std) \downarrow$ | $R(std) \uparrow$ | $e_{RMSE}(std) \downarrow$ | $R(std) \uparrow$ |
| Linear regression | 1.293(0.035) | 0.412(0.005) | 1.648(0.131) | 0.322(0.067) | 1.374(0.020) | 0.538(0.010) |
| Decision tree | 1.190(0.049) | 0.662(0.007) | 1.369(0.078) | 0.716(0.032) | 1.321(0.018) | 0.586(0.008) |
| DeepDTA | 0.792(0.089) | 0.887(0.002) | 1.312(0.139) | 0.754(0.065) | 0.904(0.019) | 0.839(0.004) |
| GraphDTA | 0.625(0.008) | <u>0.918(0.002)</u> | <u>1.029(0.054)</u> | <u>0.816(0.015)</u> | <u>0.792(0.017)</u> | <u>0.866(0.011)</u> |
| AttentionDTA | 0.702(0.020) | 0.897(0.007) | 1.119(0.042) | 0.786(0.014) | 0.861(0.021) | 0.850(0.005) |
| DEAttentionDTA | 0.846(0.053) | 0.847(0.020) | 1.157(0.189) | 0.759(0.089) | 1.058(0.112) | 0.757(0.065) |
| ImageDTA | 0.600(0.053) | 0.853(0.010) | 1.595(0.157) | 0.780(0.010) | 0.896(0.030) | 0.804(0.003) |
| DynamicDTA | <u>0.611(0.006)</u> | 0.923(0.002) | 1.007(0.050) | 0.830(0.012) | 0.727(0.017) | 0.892(0.003) |

The best results are represented in bold, and the second-best results are underlined

**Fig. 2** The performance of DEAttentionDTA (a) and DynamicDTA (b) on three datasets for the prediction of DTA

leading to better generalization and performance for DTA prediction.

Figure 2 shows the predictive performance of DynamicDTA in comparison to DEAttentionDTA, using scatter plots of true versus predicted binding affinities. This comparison reveals that DynamicDTA demonstrates a tighter clustering of points along the diagonal, indicating more accurate predictions across a wider range of binding affinities. The difference in predictive performance between DynamicDTA and DEAttentionDTA stems from the unique features of each model. While DEAttentionDTA uses

dynamic word embeddings and self-attention for linear protein and ligand sequences, DynamicDTA improves this by integrating graph-based ligand representations and dynamic protein features. This allows DynamicDTA to better capture the structural complexity and flexibility of drug–target interactions, with dynamic protein descriptors from MD simulations further enhancing its ability to model protein dynamics.

3.5 Ablation Study

In this section, we performed an ablation study to evaluate the contribution of each component in the model. Specifically, we analyzed the effect of dilated convolution, dynamic descriptors, and different feature fusion strategies. These analyses allow us to quantify the importance of each component and provide deeper insights into their influence on DTA prediction.

3.5.1 Ablation on Dilated Convolution

We replaced dilated convolutions with standard convolutions to assess their role in feature extraction. In this experiment, w/o Dilated denotes the use of standard convolutions instead of dilated convolutions. The results of this ablation study are summarized in Table 3. The substitution of dilated convolutions with standard convolutions resulted in a decline in both e_{RMSE} and R metrics, highlighting the importance of dilated convolutions in capturing multi-scale features for better predictive performance.

3.5.2 Ablation on Dynamic Descriptors

We conducted a series of ablation experiments on dynamic protein descriptors to evaluate their impact on model performance. In these experiments, w/o RMSF+Gyr indicates the removal of RMSF and Gyr from DynamicDTA, while w/o SE+MM represents the exclusion of SE and MM. Similarly, w/o RMSF+Gyr+SE and w/o Gyr+SE+MM correspond to the removal of these respective sets of dynamic descriptors, allowing us to analyze their contributions to the model. Table 3 summarizes the results of this ablation study. Removing dynamic protein features led to a decline in model performance, with the impact being more pronounced when multiple features were excluded. The removal of RMSF and Gyr alone resulted in only minor performance fluctuations, suggesting that RMSF, which captures residue-level flexibility, and Gyr, which reflects overall structural compactness,

may be partially redundant. In contrast, when Gyr, SE, and MM were simultaneously removed, the model exhibited a significant decline in performance. This shows that SE, which represents the minimum TM-score between the initial and final conformations, and MM, which captures the minimum TM-score between the most structurally divergent conformations, together provide critical insights into large-scale conformational changes. The substantial performance drop upon their removal highlights the importance of these descriptors in capturing protein structural transitions and flexibility, which are crucial for accurate binding affinity prediction.

3.5.3 Ablation on Feature Fusion

To further evaluate the effectiveness of our feature fusion approach, we conducted additional experiments comparing different fusion methods, including Concat, Sum, Average, Hadamard product, and ours (TFN). In the Concat method, drug, target, and dynamic features are concatenated along the feature dimension. The Sum method aggregates these features by element-wise addition, while the Average method further normalizes the sum by dividing the number of modalities. The Hadamard product method performs element-wise multiplication of the feature representations. Each fusion method was applied before passing the combined features to the final prediction layer. As shown in Table 4, the TFN-based fusion method achieves the best performance across all datasets, with the lowest e_{RMSE} and highest R . Traditional fusion methods such as Concat and Sum perform slightly worse, suggesting that TFN better captures complex feature interactions for improved binding affinity prediction. This improvement can be attributed to TFN's ability to model higher-order correlations between features, which are often crucial for understanding intricate biomolecular interactions. Moreover, the enhanced performance across multiple datasets further demonstrates the robustness and generalization capability of the TFN approach.

Interestingly, the performance improvement observed in the K_i^* dataset was more pronounced than that in other

Table 3 Ablation study on three datasets

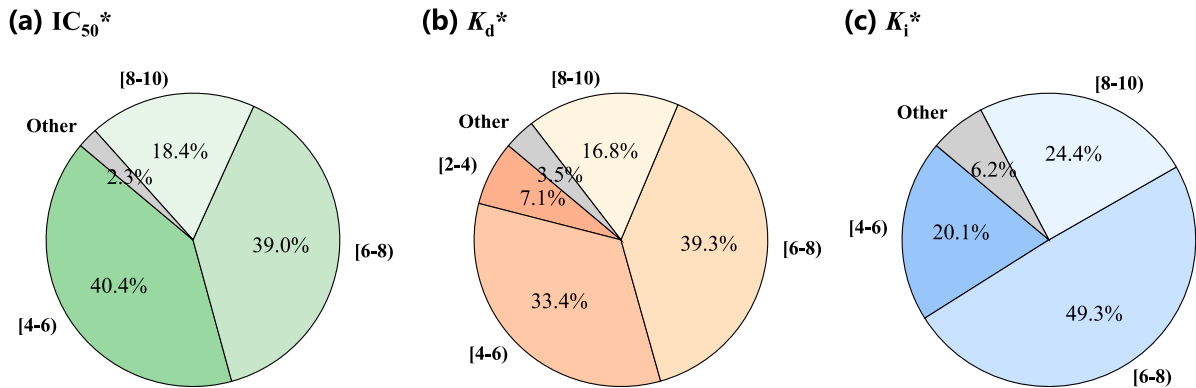
| Model | IC_{50}^* | | K_d^* | | K_i^* | |
|-------------------|--|--------------------------|--|--------------------------|--|--------------------------|
| | $e_{\text{RMSE}}(\text{std}) \downarrow$ | $R(\text{std}) \uparrow$ | $e_{\text{RMSE}}(\text{std}) \downarrow$ | $R(\text{std}) \uparrow$ | $e_{\text{RMSE}}(\text{std}) \downarrow$ | $R(\text{std}) \uparrow$ |
| w/o Dilated | 0.611(0.006) | 0.923(0.002) | 1.021(0.053) | 0.824(0.012) | 0.732(0.011) | 0.890(0.002) |
| w/o RMSF+Gyr | 0.611(0.009) | 0.923(0.003) | 1.018(0.047) | 0.826(0.011) | 0.735(0.013) | 0.889(0.002) |
| w/o SE+MM | 0.614(0.007) | 0.920(0.001) | 1.041(0.048) | 0.820(0.012) | 0.760(0.015) | 0.890(0.004) |
| w/o RMSF+Gyr+SE | 0.613(0.007) | 0.922(0.002) | 1.023(0.045) | 0.825(0.011) | 0.742(0.016) | 0.887(0.005) |
| w/o Gyr+SE+MM | 0.613(0.005) | 0.922(0.001) | 1.031(0.051) | 0.821(0.015) | 0.751(0.019) | 0.884(0.005) |
| DynamicDTA | 0.611(0.006) | 0.923(0.002) | 1.007(0.050) | 0.830(0.012) | 0.727(0.017) | 0.892(0.003) |

The best results are represented in bold

Table 4 Comparison of feature fusion methods on three datasets

| Fusion method | IC_{50}^* | | K_d^* | | K_i^* | |
|------------------|----------------------------|---------------------|----------------------------|---------------------|----------------------------|---------------------|
| | $e_{RMSE}(std) \downarrow$ | $R(std) \uparrow$ | $e_{RMSE}(std) \downarrow$ | $R(std) \uparrow$ | $e_{RMSE}(std) \downarrow$ | $R(std) \uparrow$ |
| Concat | 0.618(0.007) | <u>0.921(0.002)</u> | <u>1.008(0.044)</u> | <u>0.825(0.010)</u> | 0.763(0.023) | 0.880(0.005) |
| Sum | <u>0.616(0.010)</u> | 0.920(0.003) | 1.144(0.040) | 0.813(0.015) | 0.751(0.010) | <u>0.883(0.012)</u> |
| Average | 0.618(0.005) | 0.919(0.002) | 1.122(0.010) | 0.823(0.030) | 0.793(0.019) | 0.866(0.010) |
| Hadamard product | 0.620(0.006) | 0.920(0.005) | 1.115(0.023) | 0.821(0.020) | <u>0.730(0.015)</u> | 0.877(0.007) |
| Ours | 0.611(0.006) | 0.923(0.002) | 1.007(0.050) | 0.830(0.012) | 0.727(0.017) | 0.892(0.003) |

The best results are represented in bold, and the second-best results are underlined

**Fig. 3** Affinity distribution comparison across datasets: **a** IC_{50}^* dataset. **b** K_d^* dataset. **c** K_i^* dataset

datasets, both in the ablation and comparison experiments. This can be attributed to the inherent characteristics of the dataset. As shown in Fig. 3, the affinity distribution across the three datasets further highlights this difference. Specifically, the binding affinities in the K_i^* dataset are more concentrated, which simplifies the prediction task. This concentration enables the model to generalize better, making it more responsive to the contributions of each component.

3.6 Parameter Sensitivity Analysis

In this section, we systematically investigate the influence of four critical hyperparameters in our DynamicDTA model: the dropout rate P of the regularization, the dilation rate D of the dilated convolution, the number of attention heads H in the cross-attention layer, and the number of GCN hidden layers L for ligands.

We first examine the effect of the dropout rate P , which controls the regularization strength to prevent overfitting. We evaluated P values from the set $\{0.1, 0.2, 0.3, 0.4\}$. As shown in Fig. 4a, the model achieves the best performance when $P = 0.2$. A lower dropout rate may lead to overfitting, while a higher dropout rate may excessively reduce the model's capacity.

Next, we analyze the dilation rate D in the dilated convolution module, considering values from $\{2, 4, 6, 8\}$. Figure 4b shows that the optimal performance is achieved at $D = 4$.

A small dilation rate may limit the receptive field, making it difficult to capture long-range dependencies, whereas an excessively large dilation rate may dilute local information.

For the cross-attention layer, we evaluate different numbers of attention heads H from the set $\{2, 4, 8, 16\}$. As illustrated in Fig. 4c, the model achieves the optimal performance when $H = 4$. Increasing H can improve the model's ability to capture diverse interactions, but an excessive number may result in overfitting or oversmoothing.

Finally, we investigate the effect of the number of GCN hidden layers L for ligand representation, testing values from $\{1, 3, 5, 7\}$. Figure 4d indicates that the best performance is achieved when $L = 3$. A shallow network may lack the capacity to extract complex patterns, while an overly deep network may suffer from gradient vanishing and overfitting.

3.7 External Dataset for Validation

To further evaluate the generalization capability of our DynamicDTA model, we apply the trained model directly to the Kiba* dataset without fine-tuning. Since we performed five-fold cross-validation, we tested the best-performing model from each fold on the Kiba* dataset, and the final performance is reported as the average of these results. Table 5 presents the performance comparison between our model and baseline methods in terms of e_{RMSE} and R . The results show that DynamicDTA outperforms the baseline

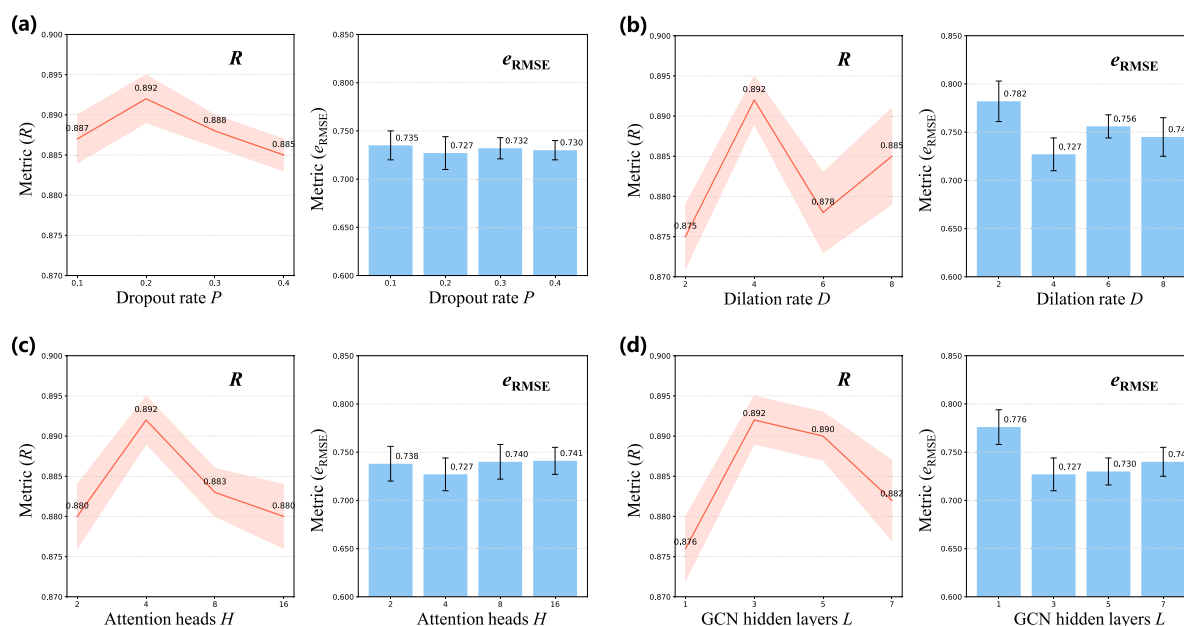


Fig. 4 Parameter sensitivity analysis in the K_i^* dataset: **a** Dropout rate P . **b** Dilation rate D . **c** Attention heads H . **d** GCN hidden layers L

Table 5 Generalization performance comparison on the Kiba* dataset

| Model | $e_{\text{RMSE}}(\text{std}) \downarrow$ | $R(\text{std}) \uparrow$ |
|-------------------|--|--------------------------|
| Liner regression | 6.615(0.020) | − 0.056(0.008) |
| Decision tree | 5.636(0.015) | − 0.071(0.006) |
| DeepDTA | 6.212(0.010) | 0.072(0.003) |
| GraphDTA | 5.954(0.023) | <u>0.176(0.007)</u> |
| AttentionDTA | <u>5.323(0.015)</u> | 0.093(0.005) |
| DEAttentionDTA | 5.816(0.012) | − 0.015(0.005) |
| ImageDTA | 5.599(0.008) | 0.010(0.004) |
| DynamicDTA | 5.090(0.013) | 0.202(0.002) |

The best results are represented in bold, and the second-best results are underlined

methods on the Kiba* dataset, achieving approximately a 4.3% improvement in e_{RMSE} and a 13% improvement in R over the second-best model. Note that since no fine-tuning was performed on the Kiba* dataset, the absolute performance is not very high (e.g., e_{RMSE} is around 1.0 and R is typically around 0.9 in comparison experiments, as shown in Table 2). These results further highlight the better generalization ability of the DynamicDTA model, demonstrating its robustness and effectiveness across different datasets without requiring additional fine-tuning.

3.8 Interpretation and Application

In this section, we analyze the model's interpretability by visualizing the attention weights assigned to protein residues

and conducting a case study to explore how the model makes predictions from a dynamic perspective.

3.8.1 Interpretability Analysis

Inspired by NHGNN-DTA [51] that employs attention weight visualization for interpretability analysis, we conducted a similar analysis on the DynamicDTA model. Specifically, we examined the cross-attention weights assigned to protein residues, as our model applies the attention mechanism exclusively to protein features, integrating protein sequence features and dynamic descriptors. Since the drug features are processed without an attention mechanism, we visualized only the attention weights associated with protein residues to identify critical binding regions prioritized by the model during DTA prediction. For this analysis, the 2FOS drug–target complex from the RCSB Protein Data Bank (RCSB PDB) was selected as a representative case. This crystallographically resolved structure contains two distinct ligand-binding sites, providing an ideal framework for validating the model's interpretability.

As illustrated in Fig. 5, the 2FOS complex exhibits two ligand-binding pockets, labeled (1) and (2) in the schematic representation. The model accurately localized residues within these functional regions. In the attention map, the top 20 residues with the highest attention weights are color-coded: red denotes correctly identified binding residues, while cyan indicates residues with erroneously elevated attention weights. Notably, the high-attention regions (red) exhibit strong spatial overlap with experimentally validated binding sites, demonstrating the model's capability to

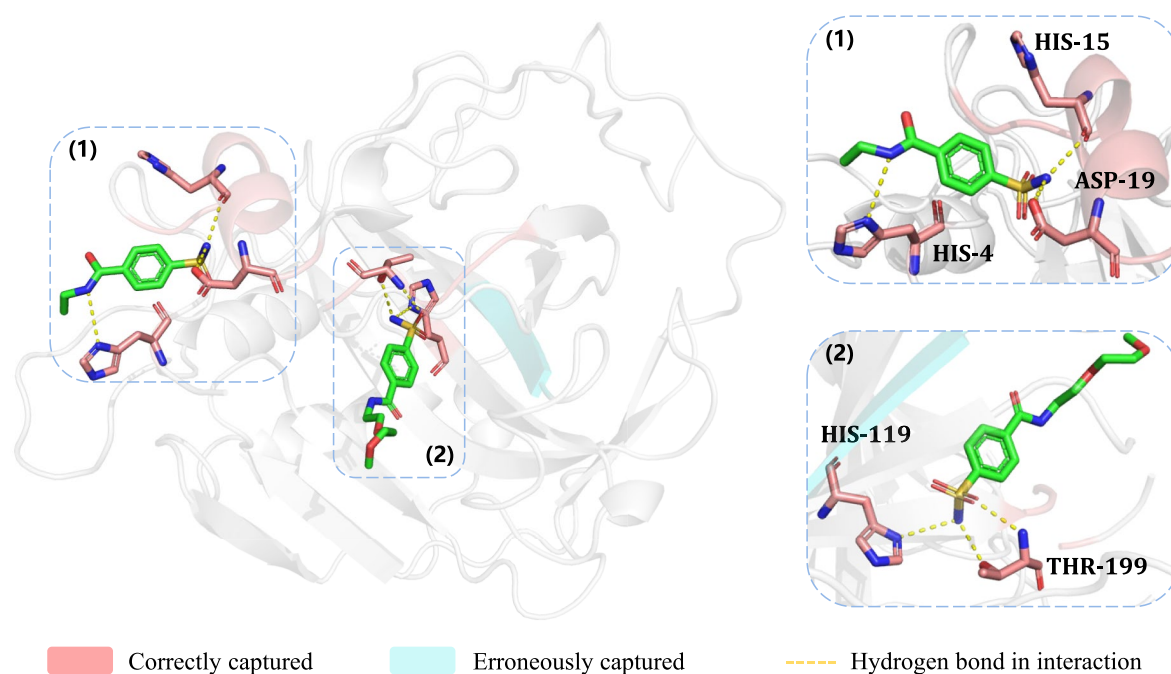


Fig. 5 Visualization of drug–target interactions in the 2FOS complex, which consists of two distinct ligand-binding regions. The model’s attention weights for the top 20 residues are highlighted, where correctly identified binding residues are shown in red, and misidentified

residues are shown in cyan. The two smaller panels on the right provide zoomed-in views of the two distinct ligand-binding regions from the left panel. These views are rotated to optimal angles to offer a clearer perspective of the binding interactions

effectively identify pharmacologically relevant interactions. A detailed examination of binding site (1) reveals that the model successfully recognizes three key interacting residues: HIS-4, HIS-15, and ASP-19 (highlighted in red). These residues are known from crystallographic studies to form hydrogen bonds and electrostatic interactions with the ligand, a finding consistent with the model’s attention patterns. This alignment between computational predictions and experimental structural data reinforces the biological plausibility of DynamicDTA’s decision-making process.

This analysis provides insights into how the DynamicDTA model focuses on key residues involved in ligand binding and highlights the model’s ability to predict these critical interactions. The inclusion of attention weight visualization further strengthens the model’s interpretability and aids in understanding the underlying mechanisms of drug–target interactions.

3.8.2 Case Study

In case study, we applied the DynamicDTA model to predict potential drugs that target *Human Immunodeficiency Virus Type 1* (HIV-1, PDB ID: 4JMU). We first extracted dynamic features of this target from our dataset and paired them with drug–target pairs from the DrugBank database. These pairs were subsequently input into the model to predict the affinity

Table 6 The top 10 predicted potential drugs targeting HIV-1

| Rank | Drug name | DrugBank ID | Evidence |
|------|--------------------------|----------------|-----------------------|
| 1 | L-methionine (R)-S-oxide | DB02235 | Unconfirmed |
| 2 | Procainamide | DB01035 | Unconfirmed |
| 3 | Oseltamivir | DB00198 | PMID: 36067538 |
| 4 | Hypochlorite | DB11123 | PMID: 10773730 |
| 5 | Tromethamine | DB03754 | PMID: 34602806 |
| 6 | Isoleucine | DB00167 | PMID: 34454514 |
| 7 | Bendazac | DB13501 | Unconfirmed |
| 8 | 3-Bromo-7-Nitroindazole | DB01997 | Unconfirmed |
| 9 | Deferiprone | DB08826 | PMID: 27191165 |
| 10 | Formaldehyde | DB03843 | PMID: 37632035 |

The drugs with related evidence found in PubMed are represented in bold

between each drug–target protein pair. After ranking the drugs based on their predicted affinity, we cross-checked the top candidates with PubMed¹ to gather supporting evidence regarding their potential therapeutic effects against HIV-1. The results, presented in Table 6, list the top 10 predicted drugs. These findings highlight the potential of our model in accelerating drug discovery.

¹ <https://pubmed.ncbi.nlm.nih.gov/>.

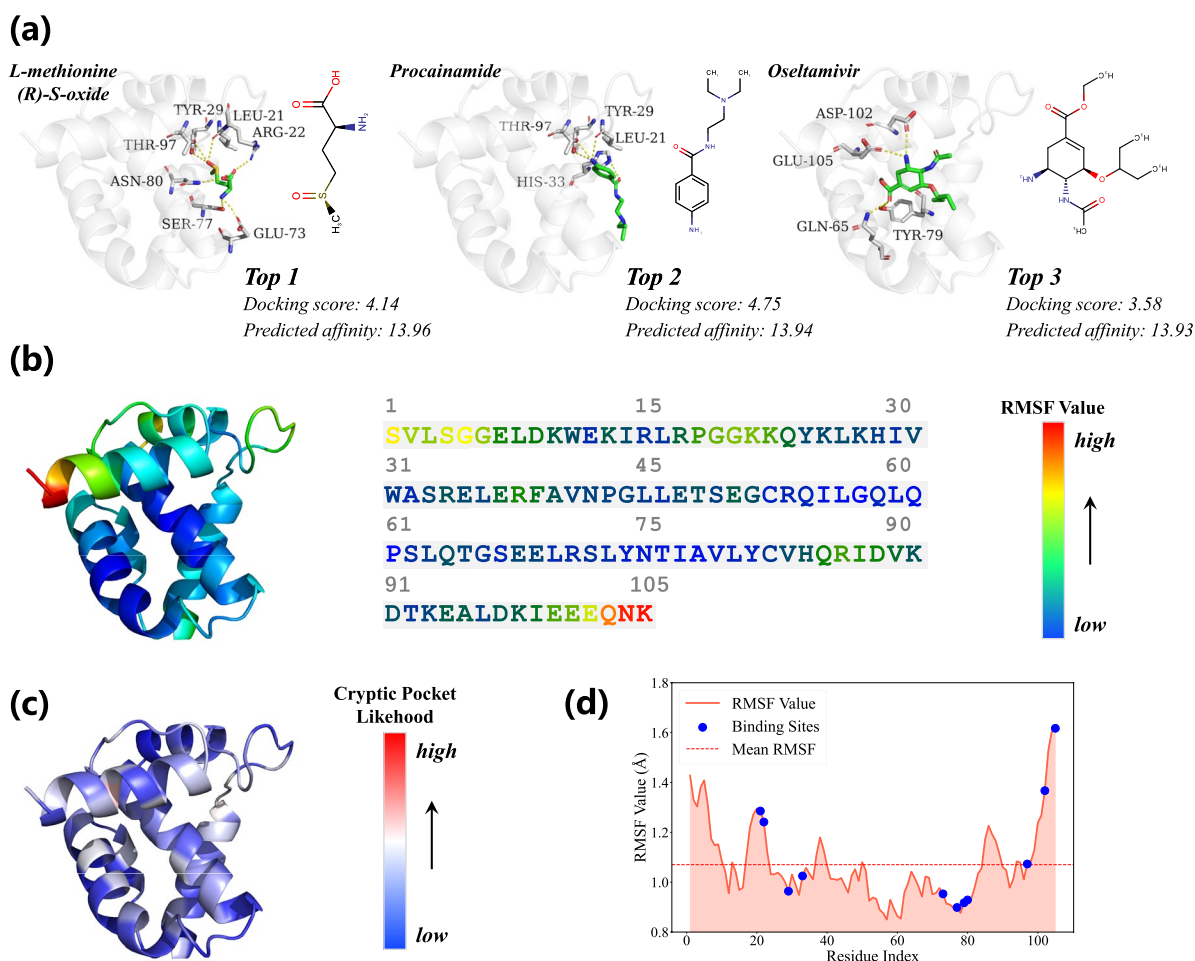


Fig. 6 Comprehensive analysis of drug-protein interactions and dynamic properties of the target protein. **a** Docking of the top three predicted drug candidates with the HIV-1 target protein, highlighting key binding residues. **b** Visualization of residue-level RMSF values mapped onto the 3D protein structure (left) and sequence (right), illustrating rigid and flexible regions. **c** PocketMiner-predicted likeli-

hood of each residue contributing to a cryptic binding pocket, with higher values indicating potential transient binding sites. **d** Line plot of RMSF values along the residue index, with blue dots marking residues involved in drug binding, demonstrating the correlation between binding sites and dynamic properties

To gain deeper insights into the interactions between the predicted drugs and the target protein, we performed a series of visualizations and analyses. First, we utilized the DeepMice² online molecular docking [52] web server to dock the top three predicted candidate drugs with the target protein, generating corresponding drug-protein complexes. These complexes were subsequently visualized using PyMOL,³ as shown in Fig. 6a. These visualizations illustrate the residue regions involved in the interactions between each drug and the HIV-1 target. Notably, Fig. 6a presents both the docking score and the predicted binding affinity. It is important to emphasize that these two metrics reflect different aspects

of the drug–target interaction. The docking score, shown in the figure, is already relatively high, indicating strong binding potential, while the predicted binding affinity provides a quantitative estimate of the actual interaction strength.

We further analyzed the dynamic properties of the target protein at the residue level. Specifically, we visualized the RMSF values of each residue, mapping the values to a color gradient to highlight regions of varying flexibility. As shown in Fig. 6b, the left panel presents the 3D structure of the protein visualized in PyMOL, while the right panel depicts the sequence-based visualization of RMSF values. Notably, some residues involved in drug binding, such as SER-77, exhibit low RMSF values, suggesting that these regions are rigid concave surfaces, which are commonly observed binding sites for small-molecule ligands. In contrast, residues like LEU-21 display higher RMSF values, indicating

² <http://www.deepmice.com/>.

³ <https://www.pymol.org/>.

increased flexibility. We hypothesize that these high-RMSF regions may correspond to cryptic binding sites [53], which are transient pockets that emerge due to protein conformational changes. To validate this hypothesis, we employed the PocketMiner [54] model, a specialized tool for predicting the likelihood of individual residues contributing to cryptic pocket formation. The results, shown in Fig. 6c, illustrate a strong correlation between high RMSF values and an increased likelihood of forming cryptic pockets. However, it is worth noting that some residues with high RMSF values do not significantly contribute to pocket formation, likely due to their inherent instability, which prevents stable drug binding.

Furthermore, Fig. 6d presents a line plot of RMSF values across the residue index, with blue dots highlighting the binding residues identified in the molecular docking study (Fig. 6a). This visualization clearly demonstrates that certain binding residues reside in low-RMSF regions, reinforcing the idea that stable, rigid concave surfaces often serve as primary binding sites. Meanwhile, other binding residues are located in high-RMSF regions, corresponding to cryptic pockets.

Overall, this case study demonstrates that our DynamicDTA model effectively captures the dynamic properties of target proteins, offering valuable insights into potential drug-binding regions and accelerating the drug discovery process.

4 Discussion and Conclusion

Accurately predicting drug–target binding affinity (DTA) is crucial in drug discovery. Existing methods often rely on static protein and drug representations, neglecting the dynamic nature of molecular interactions. To address this, we propose DynamicDTA, a framework that integrates dynamic protein features with graph-based drug representations to model complex drug–target interactions. Specifically, a cross-attention mechanism is introduced to capture both static and dynamic protein features, while a TFN seamlessly integrates multi-modal data from both drugs and proteins. Comparison experiments on three datasets demonstrate that DynamicDTA outperforms state-of-the-art methods. Moreover, a case study predicting potential drugs for HIV-1 further illustrates the model's capability in expediting the drug discovery process. However, we acknowledge several limitations in this study. First, the reliance on MD-derived dynamic descriptors may limit the model's applicability to real world where MD simulations are computationally expensive, and comprehensive MD data are not always available. Second, inconsistencies in experimental binding affinity measurements across datasets could introduce noise, affecting model performance and generalizability.

Addressing these issues will be crucial for future improvements. To overcome these limitations, in future work, we plan to explore generative AI-based approaches, such as diffusion models [55, 56], to simulate MD data for proteins that lack experimental structures, reducing the computational burden while preserving essential dynamic information.

Acknowledgements Thank the editor and anonymous reviewers for their valuable comments and suggestions. The work is supported in part by the National Natural Science Foundation of China (No. 62202413), in part by the Research Foundation of Education Bureau of Hunan Province, China (No. 23B0129), and in part by the Student Science and Technology Innovation Activity Plan of Hunan Province (No. S202410530287).

Data Availability Data is openly available in a public repository. The protein dynamic descriptors data are available at <https://doi.org/10.1093/nar/gkad1084>, and the affinity data can be found in <https://www.bindingdb.org/> and <https://www.kaggle.com/datasets/christang0002/davis-and-kiba>. All codes are available at <https://github.com/shmilyld/DynamicDTA> or from the corresponding author upon reasonable request.

Declarations

Conflict of Interest The authors have no conflict of interest relevant to the content of this article to declare.

References

1. Le NQK (2023) Predicting emerging drug interactions using GNNs. *Nat Comput Sci* 3(12):1007–1008. <https://doi.org/10.1038/s43588-023-00555-7>
2. Zhang H, Liu X, Cheng W et al (2024) Prediction of drug–target binding affinity based on deep learning models. *Comput Biol Med* 174:108435. <https://doi.org/10.1016/j.compbimed.2024.108435>
3. Tang X, Ma W, Yang M et al (2024) MFF-DTA: multi-scale feature fusion for drug–target affinity prediction. *Methods* 231:1–7. <https://doi.org/10.1016/j.ymeth.2024.08.008>
4. D'Souza S, Prema K, Balaji S et al (2023) Deep learning-based modeling of drug–target interaction prediction incorporating binding site information of proteins. *Interdiscip Sci Comput Life Sci* 15(2):306–315. <https://doi.org/10.1007/s12539-023-00557-z>
5. Sadybekov AV, Katritch V (2023) Computational approaches streamlining drug discovery. *Nature* 616(7958):673–685. <https://doi.org/10.1038/s41586-023-05905-z>
6. Tanoori B, Jahromi MZ, Mansoori EG (2021) Drug–target continuous binding affinity prediction using multiple sources of information. *Expert Syst Appl* 186:115810. <https://doi.org/10.1016/j.eswa.2021.115810>
7. Pei Q, Wu L, Zhu J et al (2023) Breaking the barriers of data scarcity in drug–target affinity prediction. *Brief Bioinform* 24(6):bbad386. <https://doi.org/10.1093/bib/bbad386>
8. Zhu Z, Zheng X, Qi G et al (2024) Drug–target binding affinity prediction model based on multi-scale diffusion and interactive learning. *Expert Syst Appl* 255:124647. <https://doi.org/10.1016/j.eswa.2024.124647>
9. Pan X, Lin X, Cao D et al (2022) Deep learning for drug repurposing: methods, databases, and applications. *Wiley Interdiscip Rev Comput Mol Sci* 12(4):e1597. <https://doi.org/10.1002/wcms.1597>

10. Zhang X, Li Y, Wang J et al (2023) A multi-perspective model for protein-ligand-binding affinity prediction. *Interdiscip Sci Comput Life Sci* 15(4):696–709. <https://doi.org/10.1007/s12539-023-00582-y>
11. 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(3):735–743. <https://doi.org/10.1021/ci400709d>
12. Davis MI, Hunt JP, Herrgard S et al (2011) Comprehensive analysis of kinase inhibitor selectivity. *Nat Biotechnol* 29(11):1046–1051. <https://doi.org/10.1038/nbt.1990>
13. Vo TH, Nguyen NTK, Le NQK (2023) Improved prediction of drug–drug interactions using ensemble deep neural networks. *Med Drug Discov* 17:100149. <https://doi.org/10.1016/j.medidd.2022.100149>
14. Stepniewska-Dziubinska MM, Zielenkiewicz P, Siedlecki P (2018) Development and evaluation of a deep learning model for protein-ligand binding affinity prediction. *Bioinformatics* 34(21):3666–3674. <https://doi.org/10.1093/bioinformatics/bty374>
15. Öztürk H, Özgür A, Ozkirimli E (2018) DeepDTA: deep drug-target binding affinity prediction. *Bioinformatics* 34(17):i821–i829. <https://doi.org/10.1093/bioinformatics/bty593>
16. Öztürk H, Ozkirimli E, Özgür A (2019) WideDTA: prediction of drug-target binding affinity. *arXiv*. <https://doi.org/10.48550/arXiv.1902.04166>
17. Zhao Q, Duan G, Yang M et al (2022) AttentionDTA: drug-target binding affinity prediction by sequence-based deep learning with attention mechanism. *IEEE/ACM Trans Comput Biol Bioinf* 20(2):852–863. <https://doi.org/10.1109/TCBB.2022.3170365>
18. Chen X, Huang J, Shen T et al (2024) DEAttentionDTA: protein-ligand binding affinity prediction based on dynamic embedding and self-attention. *Bioinformatics* 40(6):btac319. <https://doi.org/10.1093/bioinformatics/btae319>
19. Li G, Yuan Y, Zhang R (2024) Predicting protein-ligand binding affinity using fusion model of spatial-temporal graph neural network and 3d structure-based complex graph. *Interdiscip Sci Comput Life Sci* 17:1–20. <https://doi.org/10.1007/s12539-024-00644-9>
20. Nguyen T, Le H, Quinn TP et al (2021) GraphDTA: predicting drug-target binding affinity with graph neural networks. *Bioinformatics* 37(8):1140–1147. <https://doi.org/10.1093/bioinformatics/btaa921>
21. Jiang M, Li Z, Zhang S et al (2020) Drug-target affinity prediction using graph neural network and contact maps. *RSC Adv* 10(35):20701–20712. <https://doi.org/10.1039/D0RA02297G>
22. Han L, Kang L, Guo Q (2024) ImageDTA: a simple model for drug-target binding affinity prediction. *ACS Omega* 9(26):28485–28493. <https://doi.org/10.1021/acsomega.4c02308>
23. Jumper J, Evans R, Pritzel A et al (2021) Highly accurate protein structure prediction with alphafold. *Nature* 596(7873):583–589. <https://doi.org/10.1038/s41586-021-03819-2>
24. Protein Data Bank (2019) The single global archive for 3D macromolecular structure data. *Nucl Acids Res* 47(D1):D520–D528. <https://doi.org/10.1093/nar/gky949>
25. Chu Z, Huang F, Fu H et al (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>
26. Wang S, Liu Y, Zhang Y et al (2024) CHL-DTI: a novel high-low order information convergence framework for effective drug-target interaction prediction. *Interdiscip Sci Comput Life Sci* 16:1–11. <https://doi.org/10.1007/s12539-024-00608-z>
27. Lu W, Zhang J, Huang W et al (2024) DynamicBind: predicting ligand-specific protein-ligand complex structure with a deep equivariant generative model. *Nat Commun* 15(1):1071. <https://doi.org/10.1038/s41467-024-45461-2>
28. Wang H (2024) Prediction of protein-ligand binding affinity via deep learning models. *Brief Bioinform* 25(2):bbae081. <https://doi.org/10.1093/bib/bbae081>
29. Hashemzadeh S, Ramezani F, Rafii-Tabar H (2019) Study of molecular mechanism of the interaction between MEK1/2 and trametinib with docking and molecular dynamic simulation. *Interdiscip Sci Comput Life Sci* 11:115–124. <https://doi.org/10.1007/s12539-018-0305-4>
30. Liu C, Wang J, Cai Z et al. (2024) Dynamic PDB: a new dataset and a SE (3) model extension by integrating dynamic behaviors and physical properties in protein structures. *arXiv*. <https://doi.org/10.48550/arXiv.2408.12413>
31. Zadeh A, Chen M, Poria S et al (2017) Tensor fusion network for multimodal sentiment analysis. *arXiv*. <https://doi.org/10.48550/arXiv.1707.07250>
32. Vander Meersche Y, Cretin G, Gheeraert A et al (2024) ATLAS: protein flexibility description from atomistic molecular dynamics simulations. *Nucleic Acids Res* 52(D1):D384–D392. <https://doi.org/10.1093/nar/gkad1084>
33. Song X, Bao L, Feng C et al (2024) Accurate prediction of protein structural flexibility by deep learning integrating intricate atomic structures and Cryo-EM density information. *Nat Commun* 15(1):5538. <https://doi.org/10.1038/s41467-024-49858-x>
34. Henzler-Wildman K, Kern D (2007) Dynamic personalities of proteins. *Nature* 450(7172):964–972. <https://doi.org/10.1038/nature06522>
35. Ghahremanian S, Rashidi MM, Raeisi K et al (2022) Molecular dynamics simulation approach for discovering potential inhibitors against SARS-CoV-2: a structural review. *J Mol Liq* 354:118901. <https://doi.org/10.1016/j.molliq.2022.118901>
36. Lobanov MY, Bogatyreva N, Galzitskaya O (2008) Radius of gyration as an indicator of protein structure compactness. *Mol Biol* 42:623–628. <https://doi.org/10.1134/S0026893308040195>
37. Zhang Y, Skolnick J (2005) TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res* 33(7):2302–2309. <https://doi.org/10.1093/nar/gki524>
38. Lindorff-Larsen K, Piana S, Dror RO et al (2011) How fast-folding proteins fold. *Science* 334(6055):517–520. <https://doi.org/10.1126/science.1208351>
39. Raval A, Piana S, Eastwood MP et al (2012) Refinement of protein structure homology models via long, all-atom molecular dynamics simulations. *Proteins Struct Funct Bioinf* 80(8):2071–2079. <https://doi.org/10.1002/prot.24098>
40. Rivalta I, Sultan MM, Lee NS et al (2012) Allosteric pathways in imidazole glycerol phosphate synthase. *Proc Natl Acad Sci* 109(22):E1428–E1436. <https://doi.org/10.1073/pnas.1120536109>
41. Liu T, Lin Y, Wen X et al (2007) BindingDB: a web-accessible database of experimentally determined protein-ligand binding affinities. *Nucleic Acids Res* 35(suppl-1):D198–D201. <https://doi.org/10.1093/nar/gkl999>
42. Hua Y, Song X, Feng Z et al (2023) MFR-DTA: a multi-functional and robust model for predicting drug-target binding affinity and region. *Bioinformatics* 39(2):btad056. <https://doi.org/10.1093/bioinformatics/btad056>
43. Weininger D (1988) SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J Chem Inf Comput Sci* 28(1):31–36. <https://doi.org/10.1021/ci00057a005>
44. Bento AP, Hersey A, Félix E et al (2020) An open source chemical structure curation pipeline using RDKit. *J Cheminform* 12:1–16. <https://doi.org/10.1186/s13321-020-00456-1>
45. Yu F, Koltun V (2015) Multi-scale context aggregation by dilated convolutions. *arXiv*. <https://doi.org/10.48550/arXiv.1511.07122>
46. Jin Z, Wu T, Chen T et al (2023) CAPLA: improved prediction of protein-ligand binding affinity by a deep learning approach based

- on a cross-attention mechanism. *Bioinformatics* 39(2):btad049. <https://doi.org/10.1093/bioinformatics/btad049>
47. Zhang L, Wang CC, Chen X (2022) Predicting drug-target binding affinity through molecule representation block based on multi-head attention and skip connection. *Brief Bioinform* 23(6):bbac468. <https://doi.org/10.1093/bib/bbac468>
 48. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. *arXiv*. <https://doi.org/10.48550/arXiv.1412.6980>
 49. Barbet J, Huclier-Markai S (2019) Equilibrium, affinity, dissociation constants, IC₅₀: facts and fantasies. *Pharm Stat* 18(5):513–525. <https://doi.org/10.1002/pst.1943>
 50. Lagunin AA, Romanova MA, Zadorozhny AD et al (2018) Comparison of quantitative and qualitative (Q) SAR models created for the prediction of K_i and IC₅₀ values of antitarget inhibitors. *Front Pharmacol* 9:1136. <https://doi.org/10.3389/fphar.2018.01136>
 51. He H, Chen G, Chen CYC (2023) NHGNN-DTA: a node-adaptive hybrid graph neural network for interpretable drug-target binding affinity prediction. *Bioinformatics* 39(6):btad355. <https://doi.org/10.1093/bioinformatics/btad355>
 52. Li J, Fu A, Zhang L (2019) An overview of scoring functions used for protein-ligand interactions in molecular docking. *Interdiscip Sci Comput Life Sci* 11:320–328. <https://doi.org/10.1007/s12539-019-00327-w>
 53. Škrhák V, Novotný M, Feidakis CP et al (2025) CryptoBench: cryptic protein-ligand binding sites dataset and benchmark. *Bioinformatics* 41(1):btae745. <https://doi.org/10.1093/bioinformatics/btae745>
 54. Meller A, Ward M, Borowsky J et al (2023) Predicting locations of cryptic pockets from single protein structures using the PocketMiner graph neural network. *Nat Commun* 14(1):1177–1177. <https://doi.org/10.1038/s41467-023-36699-3>
 55. Corso G, Stärk H, Jing B et al (2022) Diffdock: diffusion steps, twists, and turns for molecular docking. *arXiv*. <https://doi.org/10.48550/arXiv.2210.01776>
 56. Schneuing A, Harris C, Du Y et al (2024) Structure-based drug design with equivariant diffusion models. *Nat Comput Sci* 4(12):899–909. <https://doi.org/10.1038/s43588-024-00737-x>

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

Authors and Affiliations

Dan Luo¹ · Jinyu Zhou¹ · Le Xu¹ · Sisi Yuan² · Xuan Lin^{1,3} 

✉ Sisi Yuan
syuan4@charlotte.edu

✉ Xuan Lin
jack_lin@xtu.edu.cn

¹ School of Computer Science, Xiangtan University, Xiangtan 411105, China

² Department of Bioinformatics and Genomics, the University of North Carolina at Charlotte, Charlotte 28223, USA

³ Key Laboratory of Intelligent Computing and Information Processing of Ministry of Education, Xiangtan University, Xiangtan 411105, China