ACS OMEGA

Open Access

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

Article

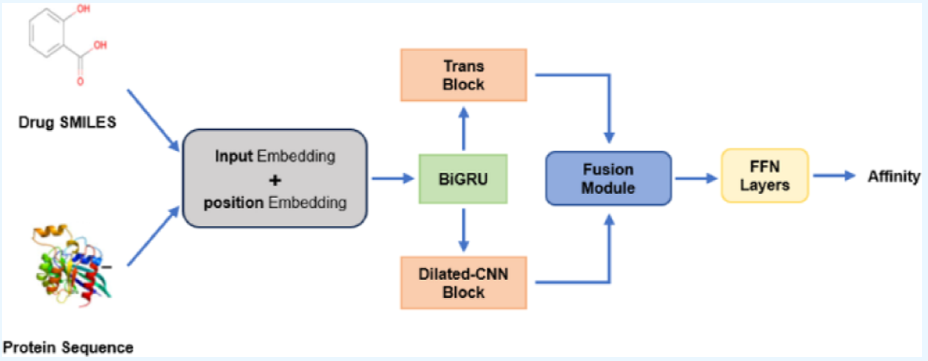
Deep Drug–Target Binding Affinity Prediction Base on Multiple Feature Extraction and Fusion

Zepeng Li, Yuni Zeng,* Mingfeng Jiang, and Bo Wei

Cite This: *ACS Omega* 2025, 10, 2020−2032

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information



ABSTRACT: Accurate drug−target binding affinity (DTA) prediction is crucial in drug discovery. Recently, deep learning methods for DTA prediction have made significant progress. However, there are still two challenges: (1) recent models always ignore the correlations in drug and target data in the drug/target representation process and (2) the interaction learning of drug−target pairs always is by simple concatenation, which is insufficient to explore their fusion. To overcome these challenges, we propose an end-to-end sequence-based model called BTDHDTA. In the feature extraction process, the bidirectional gated recurrent unit (GRU), transformer encoder, and dilated convolution are employed to extract global, local, and their correlation patterns of drug and target input. Additionally, a module combining convolutional neural networks with a Highway connection is introduced to fuse drug and protein deep features. We evaluate the performance of BTDHDTA on three benchmark data sets (Davis, KIBA, and Metz), demonstrating its superiority over several current state-of-the-art methods in key metrics such as Mean Squared Error (MSE), Concordance Index (CI), and Regression toward the mean (R_m^2). The results indicate that our method achieves a better performance in DTA prediction. In the case study, we use the BTDHDTA model to predict the binding affinities between 3137 FDA-approved drugs and severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) replication-related proteins, validating the model's effectiveness in practical scenarios.

1. INTRODUCTION

Predicting drug−target binding affinity (DTA) is essential in determining the strength of interactions between drugs and target proteins, playing a crucial role in the drug discovery process.[1−3] However, measuring DTA on a large scale purely through biological experiments is impractical. The sheer number of drug-like compounds and potential protein targets is overwhelming, and the experimental process is often time-consuming and labor-intensive.[4−6] Consequently, the use of computational methods to predict DTA scores has become necessary.

Early computational models for predicting DTA primarily utilized machine learning methods. Two notable examples are the KronRLS method,[7] based on Kronecker regularized least-squares, and SimBoost,[8] a supervised machine learning approach. While traditional machine learning methods can accurately predict DTA, they involve complex and time-consuming feature engineering, often requiring manual data processing and annotation, followed by extracting valuable features.[9] Moreover, the performance of these models tends to degrade as the data set size increases. Consequently, traditional machine learning methods have significant limitations in the field of DTA prediction.

In recent years, with the rise of deep learning and the dramatic increase in computational power, deep learning-based models have achieved great success in solving various problems in bioinformatics applications,[10,11] especially in drug discovery. Compared with traditional machine learning algorithms, deep learning offers notable advantages in handling complex

Received: September 2, 2024
Revised: December 25, 2024
Accepted: January 3, 2025
Published: January 10, 2025

nonlinear relationships and high-throughput data. Nowadays, deep learning is widely used in the field of DTA prediction, with constructed prediction methods demonstrating excellent performance. We broadly classify DTA prediction methods into two categories based on the representation of input data: sequence-based methods and graph-based methods.

Sequence-based methods mainly represent drugs and proteins as one-dimensional sequences. Typically, drugs are represented by the Simplified Molecular Input Line Entry System (SMILES), and proteins are represented by FASTA sequences. DeepDTA[12] employs numerical encoding to embed drug SMILES representations and protein sequences and then inputs them into a module consisting of a three-layer convolutional neural network (CNN) to learn the relationships of representation. AttentionDTA[13] builds on DeepDTA by improving two attention mechanisms to create a new attention method, which uses attention weights as the strength of drug−target interactions to predict DTA scores accordingly. In addition to these methods that directly perform feature learning on raw drug and protein sequences, other approaches enhance prediction accuracy by incorporating additional information or using specific algorithms to preprocess input sequences. For example, Öztürk et al.[14] designed a new CNN module based on DeepDTA, capable of extracting features from four types of data, including drug SMILES sequences and protein sequences, thereby improving predictive ability by increasing data volume and network complexity. The MGMSAN model[15] employs the BPE segmentation algorithm to create a vocabulary of high-frequency byte pairs in drug and protein sequences, efficiently learning important feature information from sequences. Kalemati et al.[16] designed a unified metric method called BiComp, which enhances protein sequence information with complementary features to improve DTA prediction quality. Although the aforementioned sequence-based methods demonstrate good performance in predicting DTA, they still have limitations in feature extraction. On one hand, increasing auxiliary information not only enriches the representation of drugs and proteins but also increases the complexity of training models and may introduce redundancy. On the other hand, using segmentation or metric algorithms to preprocess input data can cause models to focus on specific parts, leading to partial information loss.

In contrast to sequence-based methods, graph-based methods focus more on the structural information on drugs and targets. GraphDTA[17] transforms sequence information into graph structures based on various atomic features in drug sequences, utilizing four variants of Graph Neural Networks to learn drug graph representations. DGraphDTA[18] enhances prediction performance by constructing molecular graphs for drugs and building graph representations for proteins. To improve the quality of graph structure representation, TDGraphDTA[19] introduces a diffusion mechanism. Before the drug molecular graph is inserted into the graph convolution, the diffusion mechanism optimizes the graph structure representation, enabling the model to extract more meaningful and interconnected features. Graph-based methods can directly capture interactions and spatial relationships between associated elements in molecular and protein structures without considering complex relationships among other elements. However, constructing graph structures often requires extensive preprocessing, such as using external toolkits and methods to convert sequence representations into graph representations. Due to the complexity of graph structures and the lack of spatial locality, graph-based methods often require deep aggregation layers to capture information-rich local and global features, which increases computational complexity and costs.

Despite the popularity of graph-based methods in DTA prediction, research on sequence-based DTA prediction methods remains highly valuable, particularly for readily accessible raw sequence data. Given the issues identified with the aforementioned methods, our goal is to design efficient and accurate prediction methods that effectively handle the sequence information.

Moreover, existing methods for integrating drug and protein features usually use simple concatenation and attention-based interaction fusion to form the final representations.[20] For instance, DeepDTA[12] concatenates locally extracted features from two CNN blocks of drugs and proteins as the final representation for decoding. MRBDTA[21] integrates features twice to obtain the final decoded representation: first by concatenating and fusing extracted features within each modality, and second by concatenating and fusing features across both modalities. While concatenation methods are straightforward, they often overlook the complex interaction relationships between drugs and proteins. DeepCDA[22] employs a bilateral attention mechanism to generate a binding graph by merging drug and protein features, calculating the interaction strength between drug segments and protein segments. FusionDTA[23] inputs the concatenated overall features of drugs and proteins into a multihead linear attention layer to obtain aggregated features based on attention scores. AttentionMGT-DTA[24] uses a joint attention mechanism to interact cross-modal information between drugs and proteins, predicting affinity based on the interaction matrix. The interaction fusion method of MT-DTA[25] is more complex, as it captures the aggregated features of drug and protein molecules through two cascading attention mechanisms and then applies self-attention to these aggregated features to obtain the final fused representation. GPCNDTA[26] incorporates intramolecular and intermolecular cross-attention modules to interactively fuse different modal information from drugs and proteins. The intramolecular cross-attention is used to integrate different modal information related to the same biomolecule, and the intermolecular cross-attention facilitates the interaction of information between different biomolecules. Fusion methods based on attention mechanisms effectively capture interactions between drugs and proteins by computing attention weights, which roughly determine binding sites based on their magnitude. However, this computational approach often faces challenges with high-throughput data due to its complexity, and attention mechanisms can be susceptible to sample noise.

Based on our observation, there are two issues in deep models for DTA prediction: (1) Many models lack the ability to effectively combine long-range information and local features from drug and protein sequences, neglecting the importance of multiple features, and (2) oversimplified interaction learning methods are unable to deeply explore the complex interactions between drugs and targets.

To overcome the above challenges, we propose an end-to-end model named BTDHDTA for predicting DTA scores, focusing on drug and protein sequence information. Our model takes drug SMILES sequences and protein FASTA sequences as inputs. First, bidirectional gated recurrent units (GRUs) are employed to capture contextual relationships of

https://doi.org/10.1021/acsomega.4c08048
ACS Omega 2025, 10, 2020−2032

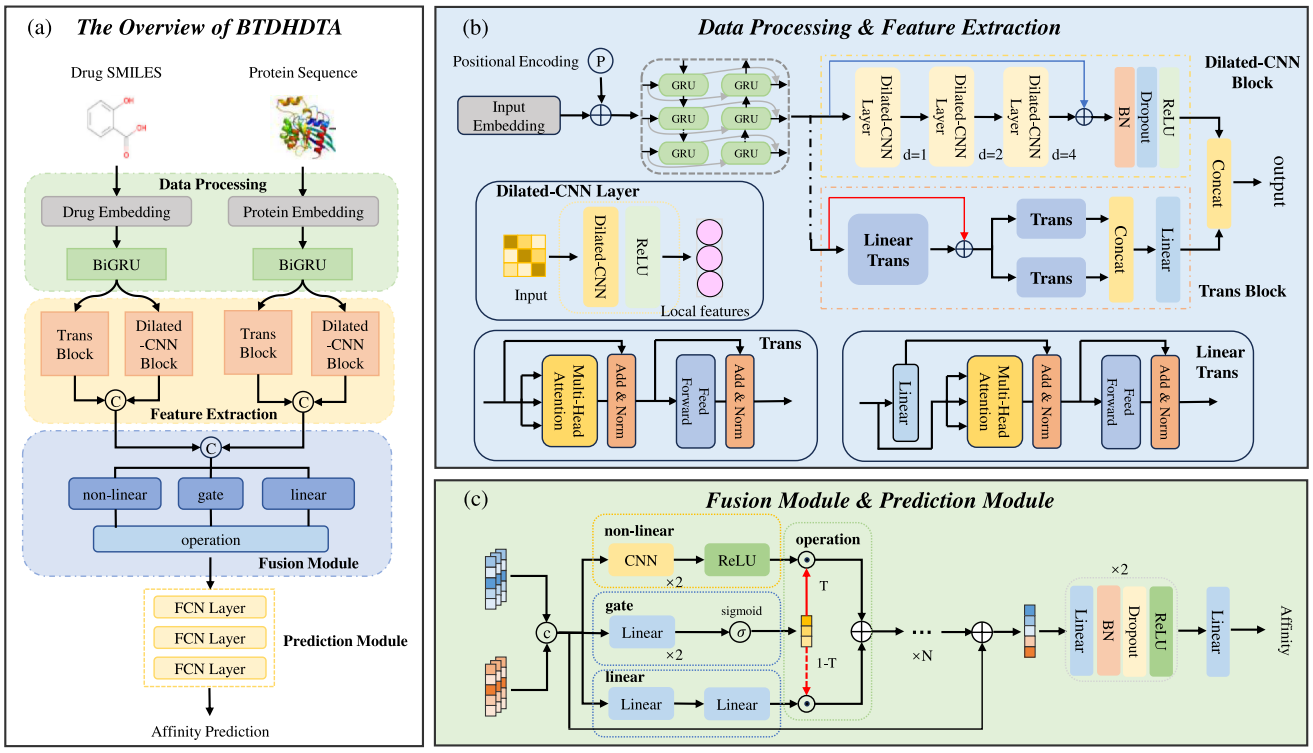


Figure 1. Schematic diagram of the BTDHDTA model proposed in this study. (a) Overall architecture of BTDHDTA. The BTDHDTA model consists of three main parts: data processing, feature extraction, and feature fusion. (b) Implementation steps for data processing and feature extraction. In the data processing stage, drug SMILES sequences and protein sequences are embedded and encoded and then passed through BiGRU to capture the temporal relationships within the sequences. In the feature extraction stage, both drug and protein sequences are passed through two blocks to extract information. The Dilated-CNN block is used to extract local information at the binding sites between drugs and proteins, while the Trans block extracts global features of the entire sequence. Here, the structures of the internal components of the two blocks are displayed. (c) Implementations of the feature fusion module and the affinity prediction module. The feature fusion module combines CNN and Highway connection to fuse drug and protein features and learn the interactions between them. The module utilizes the gating mechanism in Highway to integrate local and global features in drugs and proteins, with the CNN capturing interaction information. The prediction module consists of two feed-forward layers and one linear layer, used to decode the fused features and predict the binding scores. Here, "T" denotes the transformation gate, and "N" is set to 1.

input sequences to compensate for the shortcomings of token embedding and position embedding. Second, a Trans block and a Dilated-CNN block are parallelly integrated to extract features of the inputs. Inspired by the MRBDTA method,[21] the Trans block is used to extract global features from the entire sequence representation through improving the encoder of the transformer. The Dilated-CNN block consists of three dilated convolutions with different dilation factors and is used to capture local information at the binding sites between the drug and the target. Next, we construct a fusion module based on the Highway network to explore the interactions between drugs and proteins. The gating mechanism in the Highway network is utilized to integrate local and global features from both drugs and protein. For the transformed information determined by the gating mechanism, a convolutional neural network is used to perform nonlinear transformation and transmit the captured interaction information. Finally, the fused features are fed into fully connected neural networks (FNNs) for predicting binding affinity scores of drug−target pairs.

In the experiments, we evaluate our proposed model on three public data sets—Davis, KIBA, and Metz—and compare the model with recent sequence-based methods for DTA prediction. Our BTDHDTA model outperforms these models across all key prediction metrics. Moreover, to further validate the effectiveness of our model, we apply the model to severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) replication-related proteins, list the FDA-approved drugs with high binding affinity scores predicted by our model, and find antiviral drugs with therapeutic potential by ranking.

The main contributions of this work are as follows:

- A Bidirectional GRU (BiGRU) is built to model temporal relationships of drug/protein input sequences, incorporating dynamic temporal information into the static embedding information.
- A multiscaled feature extraction module with a Trans block and a Dilated-CNN block is proposed, which can extract global features and multiscale local features from drugs/proteins.
- A fusion module with Highway connections and CNN is proposed to integrate multigranularity drug and protein features and capture their interactions.

2. METHODS

As shown in Figure 1, we propose a novel deep learning model named BTDHDTA for predicting DTA scores. The BTDHDTA method consists of four parts: data processing, feature extraction, the fusion module, and the prediction module. Specifically, we employ BiGRU to capture the

https://doi.org/10.1021/acsomega.4c08048
ACS Omega 2025, 10, 2020−2032

contextual relationships of input sequences to make the sequences more informative after encoding drugs and proteins with token embedding and position embedding. Then, the processed sequences are separately fed into the Trans block and the Dilated-CNN block to extract global and local features. The extracted global and local features of drugs and proteins are then input into the fusion module, which combines CNN with a Highway connection, for fusion and interaction. Finally, through the prediction module, the fused representations of drugs and proteins are decoded to predict DTA scores.

2.1. Data Processing. *2.1.1. Input Embedding.* Our model takes the SMILES sequences of drugs, composed of characters representing atoms and structural indicators, and the FASTA sequences of proteins, composed of different amino acids, as input. In our study, the mathematical expression for drug D is defined as follows

$$D = d_1, d_2, ..., d_i, ..., d_{s_d}, \quad d_i \in \mathbb{N}^d \tag{1}$$

where d_i represents the ith SMILES character. N^d denotes the set containing 62 SMILES characters. The length sd of the SMILES sequence varies depending on the specific drug molecule. To ensure consistent input sizes, we define a hyperparameter l to limit the maximum length of the drug input. Inspired by transformers, we first perform token embedding for all characters in drug D. The token embedding $E^D \in R^{l\times e}$ has a trainable weight $W_t^D \in R^{v\times e}$, where v is the size of the SMILES character set and e is the embedding size of SMILES characters. To represent the relative or absolute positional relationships of each character in drug D, we apply position embedding. The position embedding $PE^D \in R^{l\times d}$ has a trainable weight $W_p^D \in R^{l\times e}$. Finally, we set the position embedding size equal to the token embedding size $(d = e)$ and add them together to obtain the output X^D of drug D

$$X^D = E^D + PE^D, \quad X^D \in \mathbb{R}^{l\times e} \tag{2}$$

Similarly, the expression for protein P is defined as follows

$$P = \{p_1, p_2, ..., p_i, ..., p_{fp}\} \quad p_i \in \mathbb{N}^p \tag{3}$$

where p_i represents the ith amino acid. N^p denotes the set of 25 common amino acids. The length fp of the protein sequence is determined by the number of amino acids that it contains. We also define a hyperparameter z as the fixed length of the protein input. X^P represents the output of protein P after token embedding and position embedding processing, defined as follows

$$X^P = E^P + PE^P, \quad X^P \in \mathbb{R}^{z\times u} \tag{4}$$

where $u(u = e)$ is the embedding size of amino acids. E^P represents the token embedding output of protein P, and PE^P represents the position embedding output.

2.1.2. Bidirectional Gated Recurrent Unit. The GRU demonstrates dynamic temporal behavior of internal network states, effectively capturing temporal information in sequence data, making it highly suitable for handling text sequence data. The built-in gating mechanism of the GRU network can solve the problem of long-range dependencies in long sequences. Although we initially add token embedding and position embedding to the sequence data, these simple embedding methods do not provide rich feature information; they merely provide static information on the sequence. To overcome this limitation, we construct a BiGRU to capture the contextual relationships of the embedded data. These contextual relationships can be fully utilized by subsequent modules, enabling the model to extract key information for prediction.

We set up two layers of BiGRU, where each layer consists of two independent GRUs: one processing the forward sequence and the other processing the backward sequence. Finally, these two sequences are concatenated to obtain output data that capture both short- and long-range dependencies. The specific steps are as follows.

We consider the embedded drug sequence X^D and protein sequence X^P as a time series. The drug sequence can be represented as $X_t^D = \{x_1, x_2, ..., x_l\}$, where $t = 1, 2, ..., l$. GRU controls the flow of information through reset and update gates. Supposing the input at time step t is x_t and the hidden state at the previous time step $t - 1$ is h_{t-1}, the formulas for the update gate z_t and the reset gate r_t are computed as follows

$$\begin{cases} z_t = \sigma(W_z x_t + U_z h_{t-1}) \\ r_t = \sigma(W_r x_t + U_r h_{t-1}) \end{cases} \tag{5}$$

where W and U are learnable weight matrices determining the influence of input x_t on outputs z_t and r_t. The function $\sigma(\cdot)$ denotes the sigmoid activation function, thus constraining the values of z_t and r_t within the range 0 to 1. Once z_t and r_t are obtained, the candidate state h_t' and the hidden state h_t at time step t are computed as follows

$$\begin{cases} h_t' = \tanh(W_x x_t + r_t \odot (U_r \cdot h_{t-1})) \\ h_t = (1 - z_t) \odot h_{t-1} + z_t \odot h_t' \end{cases} \tag{6}$$

where W_x is a learnable weight matrix, r_t controls the proportion of information from the previous state h_{t-1} to the candidate state h_t', $\tan(\cdot)$ denotes the tanh activation function, and $\odot$ represents element-wise multiplication. Similar to r_t, z_t controls the proportion of information from the previous state h_{t-1} and the candidate state h_t' to update the current hidden state h_t.

Due to BiGRU consisting of two GRUs in opposite directions, at each time step, the output is jointly determined by these two unidirectional GRUs. We express formulas 5 and 6 using the function $\mathrm{GRU}(x_t, h_{t-1})$. Therefore, the BiGRU computation formula can be expressed as

$$h_t = \mathrm{GRU}(x_t, \overrightarrow{h_{t-1}}) \| \mathrm{GRU}(x_t, \overleftarrow{h_{t-1}}) \tag{7}$$

The notation $\|$ denotes concatenation, where the left $\mathrm{GRU}(\cdot)$ represents the forward hidden state and the right one represents the backward hidden state. h_t is the final output at time step t.

2.2. Feature Extraction. *2.2.1. Dilated-CNN Block.* CNN is good at recognizing patterns through a weight-sharing strategy. By increasing the number of layers, CNN filters can capture more abstract features. However, CNNs are typically constrained by fixed-size convolutional kernels, which limits the scope of feature extraction. In contrast, the dilated convolution can expand the receptive field by setting different dilation rates while keeping the number of parameters and the size of the output feature map unchanged. This expanded receptive field allows for capturing multiscale features over a larger range of information.

We have developed a Dilated-CNN block that consists of three stacked 1D dilated convolutional layers for local feature extraction. ReLU activation functions are applied to each

https://doi.org/10.1021/acsomega.4c08048
ACS Omega 2025, 10, 2020−2032

dilated convolutional layer. The specific computation formulas are as follows

$$\mathrm{ConV}(X_l)_{i,k} = \mathrm{ReLU}\{\sum_{m=0}^{M-1} W_m^{(l)} \cdot X_{i\cdot\mathrm{step}+k_l}\} \tag{8}$$

where l is the current layer number, k is the dilation factor, M is the length of the filter, and W_m denotes the mth weight in the kernel.

In dilated convolution, the dilation factor k means that each unit in the kernel corresponds to input regions spaced apart by k units both vertically and horizontally. The receptive field varies with the value of k. For our model, we set three different values for k to control the feature extraction range for each convolutional layer. Since the number of relevant atoms near drug−protein binding sites and amino acid residues is generally limited, most around 3−5 drug atoms and 8−12 protein amino acid residues, we set the dilation factors for the three dilated convolutional layers to 1, 2, and 4, respectively. After passing through these three layers of dilated convolution, we apply normalization and drop-out to the output. Finally, residual connections are applied across the entire Dilated-CNN block.

2.2.2. Trans Block. The transformer model has been widely used to process sequence data and is good at capturing global information in sequences. Building on the work of Zhang et al.,[21] we introduce the encoder of the transformer as the foundational component of the Trans block, called Trans. Adding a linear layer at the beginning of a Trans constitutes a Linear-Trans. The Trans block consists mainly of one Linear-Trans and two parallel Trans, with residual connections between the Linear-Trans and the Trans. The detailed implementation steps of the Trans block are as follows.

For the basic component, Trans, it includes a multihead attention layer and a position-wise feed-forward layer, connected by a residual connection (Add operation) and a layer normalization (Norm operation). The multihead attention layer is the core of Trans and consists of $h(h = 4)$ parallel scaled dot-product attention layers. Scaled dot product attention is a type of generalized attention that uses queries (Q), keys (K), and values (V). It maps a query and a set of key-value pairs to an output. Specifically, the e-dimensional input of Trans, $F \in R^{n\times e}$, is set as the Q, K, and V for the scaled dot-product attention layer. Then, linear transformations are applied to Q, K, and V, respectively, to obtain h different sets of Q_i, K_i, and $V_i(i = 1, 2, 3, 4)$ as shown below

$$\begin{cases} Q_i = Q\cdot W_i^Q \\ K_i = K\cdot W_i^K \\ V_i = V\cdot W_i^V \end{cases} \tag{9}$$

where $Q_i \in R^{n\times d_k}$, $K_i \in R^{n\times d_k}$, and $V_i \in R^{n\times d_v}$. n is the maximum length l(or z) of drug X^D (or protein X^P). The output of each scaled dot-product attention layer, denoted as head_i, is calculated as follows

$$\mathrm{Attention}(Q, K, V) = \mathrm{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{10}$$

Here, during the computation of the attention matrix, we did not apply a mask operation, as the prior data processing has already assigned contextual relevance to each element in the sequence, making them all meaningful.

Finally, the outputs of h scaled dot-product attention layers are concatenated and passed through a linear layer to generate the output MH of the multihead attention layer

$$\mathrm{MH} = \mathrm{Concat}(\mathrm{head}_1, \ldots, \mathrm{head}_i, \ldots, \mathrm{head}_h)W^O \tag{11}$$

where $W^O \in R^{hd_k\times e}$ is the weight matrix for the linear layer.

The output of the multihead attention layer is fed into the position-wise feed-forward layer for a series of nonlinear transformations, after applying a residual connection and a normalization. The feed-forward layer is made up of two linear layers with a ReLU activation function in between. The main difference between Linear-Trans and Trans is that Linear-Trans inputs the linearly transformed X^P or X^D into the residual connection after the multihead attention layer, rather than directly inputting X^P or X^D. Since capturing global features from long sequences may not extract all the information when processed from a single perspective, we employ parallel Trans structures; we adopt a parallel Trans structure. The parallel Trans can process the same inputs from multiple perspectives independently to ensure the full extraction of global features. Moreover, the features captured by each Trans can complement each other, improving the robustness of the model.

At the end of the entire feature extraction module, we concatenated the global features extracted from the Trans block with the local features extracted from the Dilated-CNN block to form the final output. The output for the protein is $H^P \in R^{z\times 2e}$, and the output for the drug is $H^D \in R^{l\times 2e}$.

2.3. Feature Fusion Module. After the internal features of the protein and the drug are extracted, the next step is to capture the relational information between them. Most existing end-to-end models simply concatenate the drug and protein features and then apply a multilayer perceptron to learn their relationships. However, this fusion approach does not effectively construct the interactions between the protein and the drug, often neglecting their actual interactions.[27,28] As a result, it fails to capture the associated features effectively, which can limit the predictive capability of the model.

To better capture the interactions between proteins and drugs, we designed a feature fusion module based on the Highway network. The Highway network architecture uses gating mechanisms to regulate the flow of information, enabling smooth information passage through deep networks and mitigating the vanishing gradient problem.[29] Additionally, it efficiently integrates information on varying granularities.[30] After the feature extraction phase, we performed average pooling on the drug output H^D and the protein output H^P. Then, these pooled results are concatenated to form the input $X \in R^{4e}$ for the feature fusion module. Considering that the input matrix X combines both global and local features of the drug and protein sequences, there may be some redundancy. We utilize the carry gate (C) and transform gate (T) within the Highway network. These gates help in managing and adjusting the flow of information, thereby integrating global and local information while minimizing redundancy. Furthermore, we introduce a one-dimensional convolution operation into the Highway network to capture interaction relationships while applying nonlinear transformations to the input matrix X. The kernel size is set to 3. The detailed computation process is shown below

https://doi.org/10.1021/acsomega.4c08048
ACS Omega 2025, 10, 2020−2032

$$\begin{cases} y = H(\mathrm{Conv}(X), W_H)\cdot T(X, W_T) + X\cdot C(X, W_C) \\ T(X, W_T) = \sigma(W_T X + b_T) \end{cases} \quad (12)$$

Here, $\cdot$ denotes element-wise multiplication, W represents the learnable weight parameters, and $H(\cdot)$ is a nonlinear activation function, which is ReLU in this work. $\mathrm{Conv}(\cdot)$ is a 1D convolution function. $T(\cdot)$ is the transform gate in the Highway network, while σ denotes the Sigmoid function. The carry gate, $C(\cdot)$, is defined as $1 - T(\cdot)$ for the sake of simplicity. We employ a residual connection to add the input X to output y of the improved Highway network, obtaining the final output.

2.4. Prediction Module. The primary function of the interaction module is to predict the binding affinity score for a drug−target pair, based on the deeply integrated representations of the drug and protein. The interaction module comprises two feed-forward layers and a final linear layer. Each feed-forward layer is composed of a linear layer, followed by normalization, dropout, and a ReLU activation function. These layers are used to decode the interaction information R obtained from the feature fusion module. The final linear layer generated the predicted binding affinity score

$$y^* = W_L\cdot \mathrm{FFN}(R, 2) \quad (13)$$

where W_L represents the weight parameters in the linear layer, and y^* denotes the predicted binding affinity score of the drug−target pair.

3. RESULTS AND DISCUSSION

In this section, we experiment with our proposed model, BTDHDTA, on three public data sets—Davis, KIBA, and Metz. We use Concordance Index (CI), R_m^2, Pearson, and Spearman metrics to assess the performance of our model in comparison to baseline models. In DTA prediction, CI and R_m^2 are the most commonly used and primary evaluation metrics, while Pearson and Spearman correlation coefficients are used to assess the correlation between the predicted values and the actual values.

3.1. Benchmark Data Set. We test and evaluate our model on three publicly available benchmark data sets: the Davis kinase data set,[31] the KIBA data set,[32] and the Metz data set.[33] These data sets are widely used in prior studies for DTA prediction.

3.1.1. Davis Data Set. The Davis data set includes 30,056 interactions among 442 proteins and 68 drugs, with binding affinities represented by dissociation constant (K_d) values. To manage numerical stability and ensure appropriate scaling, we transformed the K_d values into the logarithmic domain to obtain pK_d values.[8,12] This transformation is done by computing the negative logarithm of the K_d values, as shown in the following equation

$$pK_d = -\log_{10}\left(\frac{K_d}{1e9}\right) \quad (14)$$

3.1.2. KIBA Data Set. The KIBA data set is derived using the KIBA method, which combines various inhibitor metrics, such as K_i, K_d, and IC_{50}, to compute KIBA scores. These scores quantify the biological activity of kinase inhibitors and are used as measures of the binding affinity. The KIBA data set encompasses 118,254 interactions involving 229 proteins and 2111 drugs.

3.1.3. Metz Data Set. The Metz data set is a public resource comparable in size to the Davis data set. It is widely used in the fields of deep learning and drug discovery. The Metz data set contains 1423 drugs, 170 proteins, and 35,259 interactions. Detailed information about the Davis, KIBA, and Metz data sets is summarized in Table 1.

Table 1. Summary of the Three Datasets

data set	drugs	proteins	interactions	train set	test set
Davis	68	442	30,056	25,046	5010
KIBA	2111	229	118,254	98,545	19,709
Metz	170	1423	35,259	28,207	7052

3.2. Evaluation Metrics. Since our task is a regression task, we use Mean Squared Error (MSE) as the loss function to optimize the weights of the BTDHDTA model, aiming to minimize the discrepancy between the predicted values P_i and the true values Y_i. The MSE is calculated as follows

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(P_i - Y_i)^2 \quad (15)$$

We also use the Root Mean Square Error (RMSE) to measure the prediction error of the model. Combining MSE and RMSE offers a more comprehensive and clearer error analysis.

The CI[34] is used as an evaluation metric for model performance. The CI measures the probability of consistency between the true values and predicted values. It is defined as follows

$$\mathrm{CI} = \frac{1}{Z}\sum_{\delta_i>\delta_j} h(b_i - b_j) \quad (16)$$

where b_i is the predicted value for affinity δ_i, b_j is the predicted value for affinity δ_j, Z is the normalization constant, and $h(x)$ denotes the step function[5]

$$h(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0.5, & \text{if } x = 0 \\ 0, & \text{if } x < 0 \end{cases} \quad (17)$$

To evaluate the external predictive performance of the model, we also use Regression toward the mean (R_m^2) index. This metric is commonly employed in regression-based QSAR models to evaluate how well the model predicts new, unseen data[35]

$$r_m^2 = r^2*(1 - \sqrt{r^2 - r_0^2}) \quad (18)$$

where r is the squared correlation coefficient with intercept, and r_0 is the squared correlation coefficient without intercept. A model is deemed acceptable if its R_m^2 value on the test set exceeds 0.5.

There are two metrics that measure the correlation between the true values and the predicted values, namely, the Pearson's correlation coefficient (Pearson) and the Spearman's rank correlation (Spearman). Pearson quantifies the linear correlation between two continuous variables, while Spearman measures the nonlinear correlation between two ranked variables (nonparametric).

3.3. Experimental Setup. We validated the performance of our model on the benchmark data sets Davis, KIBA, and

https://doi.org/10.1021/acsomega.4c08048
ACS Omega 2025, 10, 2020−2032

Metz. Using the same data splitting approach as GraphDTA,[17] we partitioned the entire sample space into training and test sets with a 5:1 ratio. To prevent the model from stagnating, we employed a dynamically adjusted learning rate based on the loss values during training. Based on the parameter ranges reported in the literature,[21,22,29,40] the suitable values for parameters such as convolution kernel size, the number of convolution kernels, and the number of heads in multihead attention were determined through parameter selection experiments. The Davis data set was trained on a single NVIDIA 3090 GPU, while the KIBA data set and the Metz data set were trained on a single NVIDIA 4090 GPU. Table 2 summarizes the optimal parameter settings used for model training.

Table 2. Summary of the Parameters for Davis, KIBA, and Metz Data Sets

parameter	Davis/KIBA/Metz
max length of drug	85/100/100
max length of protein	1200/1000/1200
embedding size	128
layers of BiGRU	2
heads in multihead attention	4
number of filters in Dilated-CNN	32 64 128
filter size of drug	4
filter size of protein	12
dilation rates of Dilated-CNN	1 2 4
layers of CNN or Linear in fusion module	2
layers of fusion module (N)	1
batch size	64
epoch	400/800/500
optimizer	Adam
learning rate	0.0001
activation function	ReLU

3.4. Experiment 1: BiGRU for Input Data Processing.

To address the problem that token embedding and position embedding cannot provide contextual relationships for the original input sequences, we use BiGRU after the two embedding methods. BiGRU can capture rich contextual information on input sequences instead of just static information. This enables the model to learn complex sequence features.

To assess the impact of data processing using BiGRU on model prediction performance, we conducted a controlled experiment. The MRBDTA model served as the baseline for the proposed BTDHDTA method. By incorporation of BiGRU into the baseline, an experimental model was created. Table 3 displays the performance metrics of both baseline and the experimental model on the Davis and KIBA data sets. The results indicate that adding BiGRU improved the CI metric by 0.004 on the Davis data set and by 0.007 on the KIBA data set; the R_m^2 metric increased by 0.01 on the Davis data set and by 0.007 on the KIBA data set. The Pearson and Spearman metrics both improved on the two data sets, with increases of 0.008 and 0.005 on the Davis data set and 0.011 on the KIBA data set. Additionally, the MSE metric decreased by 0.01 on the Davis data set and by 0.013 on the KIBA data set. The results show that using BiGRU to capture contextual information on the input data improves the performance of subsequent feature extraction, with a slightly better effect observed on the KIBA data set compared to the Davis data set.

3.5. Experiment 2: Effects of Dilated-CNN in the Feature Extraction Module.

Although the Trans block, comprising Linear-Trans encoders and two Trans encoders, effectively captures global features from drug and protein inputs, it may not be as effective at extracting local features, such as binding sites, as convolutional methods. Therefore, we incorporated a Dilated-CNN module into our model.

In this subsection, we evaluated whether the Dilated-CNN module, based on three layers of dilated convolution, could provide local feature information to complement the deficiencies of the Trans block in this regard. The comparison model for this subsection is the experimental model from Experiment 1, with the addition of Dilated-CNN to create the experimental model. Performance comparisons between these two models on the Davis and KIBA data sets are shown in Table 4. The experimental model outperforms the comparison model on both data sets. On the KIBA data set, despite no noticeable change in MSE, the RMSE still decreased by 0.001, the R_m^2 metric improved, and the model predicted more accurately in the sample space. This suggests that the Dilated-CNN module effectively captures local feature information from the spatial binding sites of drug−target pairs, enhancing overall feature extraction.

Furthermore, this effect becomes more pronounced with increasing data volume.

3.6. Experiment 3: Effects of Feature Fusion.

To validate the effects of the proposed drug−target fusion module, we conducted ablation experiments on the Davis and KIBA data sets and compared them with several popular fusion methods on both data sets. The results are shown in Table 5.

First, we conducted ablation experiments to evaluate the effects of our fusion module. The results show that models incorporating our fusion module outperformed those using simple concatenation methods across all metrics. This indicates that our fusion module is more adept at capturing the intricate interactions between the drug and protein features. Furthermore, we replaced our fusion module with alternative attention-based fusion methods, including Cross Attention,[36,37] Mutual Attention,[38] and the fusion module from ref 28, and then compared their performance. Our fusion method outperforms these alternatives across all metrics. In fact, the performance of these attention-based fusion methods is inferior to that of simple concatenation, suggesting that more

Table 3. Performance of BiGRU on the Davis and KIBA Data Sets[a]

data set	method	CI	R_m^2	Pearson	Spearman	RMSE	MSE
Davis	baseline	0.898	0.714	0.853	0.712	0.467	0.218
	+BiGRU	0.902	0.724	0.861	0.717	0.456	0.208
KIBA	baseline	0.887	0.773	0.887	0.879	0.380	0.145
	+BiGRU	0.894	0.780	0.898	0.890	0.363	0.132

[a]The baseline is the MRBDTA model, and the experimental model is +BiGRU.

https://doi.org/10.1021/acsomega.4c08048
ACS Omega 2025, 10, 2020−2032

Table 4. Performance of the Three-Layer Dilated Convolution Module (Dilated-CNN) on the Davis and KIBA Data Sets[a]

data set	method	CI	R_m^2	Pearson	Spearman	RMSE	MSE
Davis	baseline + BiGRU	0.902	0.724	0.861	0.717	0.456	0.208
	+Dilated-CNN	0.905	0.728	0.866	0.720	0.448	0.200
KIBA	baseline + BiGRU	0.894	0.780	0.898	0.890	0.364	0.132
	+Dilated-CNN	0.897	0.794	0.898	0.893	0.363	0.132

[a]The comparison model is the baseline + BiGRU, and the experimental model is the +Dilated-CNN.

Table 5. Performance of the Fusion Module on the Davis and KIBA Data Sets[a]

data set	method	interaction	CI	R_m^2	Pearson	Spearman	RMSE	MSE
Davis	baseline + BiGRU + Dilated-CNN	concatenation	0.905	0.728	0.866	0.720	0.448	0.200
		cross attention	0.889	0.642	0.841	0.697	0.490	0.240
		mutual attention	0.891	0.693	0.855	0.698	0.467	0.218
		fusion in ref 28	0.894	0.690	0.851	0.703	0.472	0.223
	BTDHDTA	fusion module	0.907	0.733	0.865	0.725	0.449	0.201
KIBA	baseline + BiGRU + Dilated-CNN	concatenation	0.897	0.794	0.898	0.893	0.363	0.132
		cross attention	0.881	0.696	0.881	0.870	0.396	0.157
		mutual attention	0.892	0.770	0.891	0.884	0.375	0.141
		fusion in ref 28	0.893	0.767	0.893	0.885	0.372	0.139
	BTDHDTA	fusion module	0.898	0.805	0.901	0.893	0.357	0.127

[a]The comparison model is the baseline + BiGRU + Dilated-CNN, and BTDHDTA is our complete model, which adds the Fusion module to the comparison model.

Table 6. Our Model's Test Results on the Davis Data Set Compared to Other Advanced Methods

method	CI	R_m^2	MSE	RMSE	Pearson	Spearman
DeepDTA	0.878	0.630	0.261	0.511	0.846	0.690
DeepCDA	0.891	0.649	0.248	0.498	0.857	0.716
MATT_DTI	0.891	0.683	0.227	0.526		
DeepFusionDTA	0.887		0.253	0.503		
ELECTRA-DTA	0.897	0.671	0.238	0.488	0.844	
BiComp-DTA	0.904	0.696	0.237	0.487		
SSM-DTA	0.890		0.219	0.468		
MFR-DTA	0.905	0.705	0.221	0.470		
ImageDTA	0.901		0.215	0.464		
TC-DTA	0.886	0.670	0.231	0.481		
ours	**0.907**	**0.733**	**0.201**	**0.449**	**0.865**	**0.725**

Table 7. Our Model's Test Results on the KIBA Data Set Compared to Other Advanced Methods

method	CI	R_m^2	MSE	RMSE	Pearson	Spearman
DeepDTA	0.863	0.673	0.194	0.440	0.848	0.828
DeepCDA	0.889	0.682	0.176	0.420	0.855	0.836
MATT_DTI	0.889	0.756	0.150	0.387		
DeepFusionDTA	0.876		0.176	0.420		
ELECTRA-DTA	0.889	0.727	0.162	0.402	0.879	
BiComp-DTA	0.891	0.757	0.167	0.409		
SSM-DTA	0.895		0.154	0.392		
MFR-DTA	**0.898**	0.789	0.136	0.369		
ImageDTA	0.886		0.147	0.383		
TC-DTA	0.877	0.734	0.177	0.421		
ours	**0.898**	**0.805**	**0.127**	**0.357**	**0.901**	**0.893**

complex fusion approaches may not be suitable for handling intricate and highly complementary feature representations. Since the feature vectors obtained after feature extraction contain both global and local features of two types, which are highly complementary, attention-based fusion methods might inadvertently learn unnecessary relationships during modality interaction. This further validates the superiority of our fusion module in effectively managing drug−protein interactions and handling complex feature representations.

3.7. Experiment 4: Comparison to Other Advanced Methods. For the Davis and KIBA data sets, we compared our model with ten methods (DeepDTA,[12] DeepCDA,[22] MATT_DTI,[39] DeepFusionDTA,[40] ELECTRA-DTA,[41] BiComp-DTA,[16] SSM-DTA,[42] MFR-DTA,[43] ImageDTA,[44] and TC-DTA[49]). Tables 6 and 7, respectively, list the results of these models in predicting binding affinities on the Davis and KIBA data sets. For the Metz data set, we compared our model with three methods (DeepDTA,[12] MRBDTA,[21] and Modality-

https://doi.org/10.1021/acsomega.4c08048
ACS Omega 2025, 10, 2020−2032

Table 8. Our Model's Test Results on the Metz Data Set Compared to Other Advanced Methods

method	CI	R_m^2	MSE	RMSE	Pearson	Spearman
DeepDTA	0.815	0.678	0.286	0.535	0.835	0.792
MRBDTA	0.812	0.687	0.281	0.530	0.834	0.786
modality-DTA	0.794		0.281	0.530		
ours	**0.826**	**0.724**	**0.253**	**0.503**	**0.854**	**0.813**

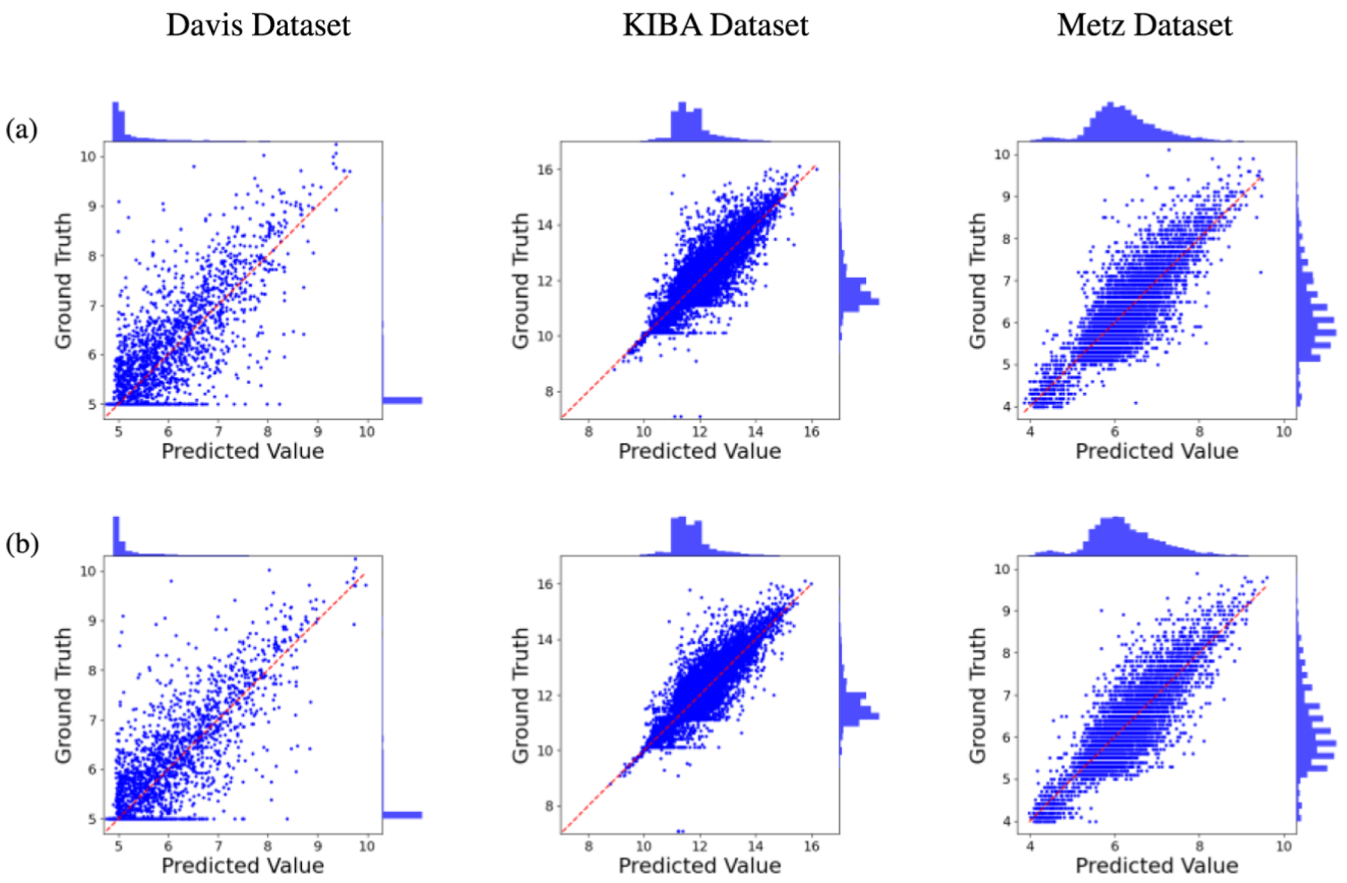


Figure 2. Visualization of the scatter plots and frequency histograms of the predicted affinity values and ground truths for (a) MRBDTA and (b) BTDHDTA methods on the Davis, KIBA, and Metz data sets. The red dashed line represents the regression line.

DTA[5]). The comparison results are listed in Table 8. These selected methods are all sequence-based methods, do not involve structural information, and are all representative methods in the field of DTA prediction.

As shown in the tables, our model outperforms most sequence-based methods. Specifically, on all data sets, our model outperforms the other comparison models in most metrics. The CI metric improved to 0.907 and 0.898, respectively, reaching the best on the Davis data set and equaled to the MFR-DTA model on the KIBA data set. The R_m^2 metric improved to 0.733 and 0.805, respectively, higher than the other models. And the Pearson metric reached above 0.9 on the KIBA data set. On the Metz data set, our model also outperforms other advanced methods across all metrics.

To statistically evaluate the significant improvement of our method, the paired *t*-test was utilized at a significant level of 0.05. On all data sets and metrics, the *P*-values of our model and the compared methods are all below 0.01, lower than the significance level of 0.05. This test indicates that our method outperforms other advanced methods. In particular, compared to the baseline method MRBDTA, the *P*-values are below 0.01 on the Davis data set and below 0.001 on the KIBA and Metz data sets, further demonstrating the significance of the performance improvement.

We also designed a visualization experiment to analyze the distribution between the predicted affinity values and ground truths of our model and compared it with the baseline MRBDTA. First, a scatter plot and a frequency histogram are used to illustrate the deviation between predicted and true values. Generally, the more points concentrated on the red dashed line $y = x$ mean the more accurate the model's predictions and the better performance of the method. Figure 2 shows the scatter plots and frequency histograms of true and predicted values for our model and the baseline MRBDTA on three data sets. The points predicted by the BTDHDTA method are more concentrated around the regression line than those of the MRBDTA method, with fewer outliers. This indicates that the BTDHDTA model has excellent predictive capability and robustness. Figure 3 shows the kernel density estimate plots of the affinity between true and predicted values. In the kernel density plots, the more the two color graphics coincide, the closer the data distribution between predicted and true values. We calculated the area of overlap for the two models, with BTDHDTA achieving overlap areas of 0.9099,

https://doi.org/10.1021/acsomega.4c08048
ACS Omega 2025, 10, 2020−2032

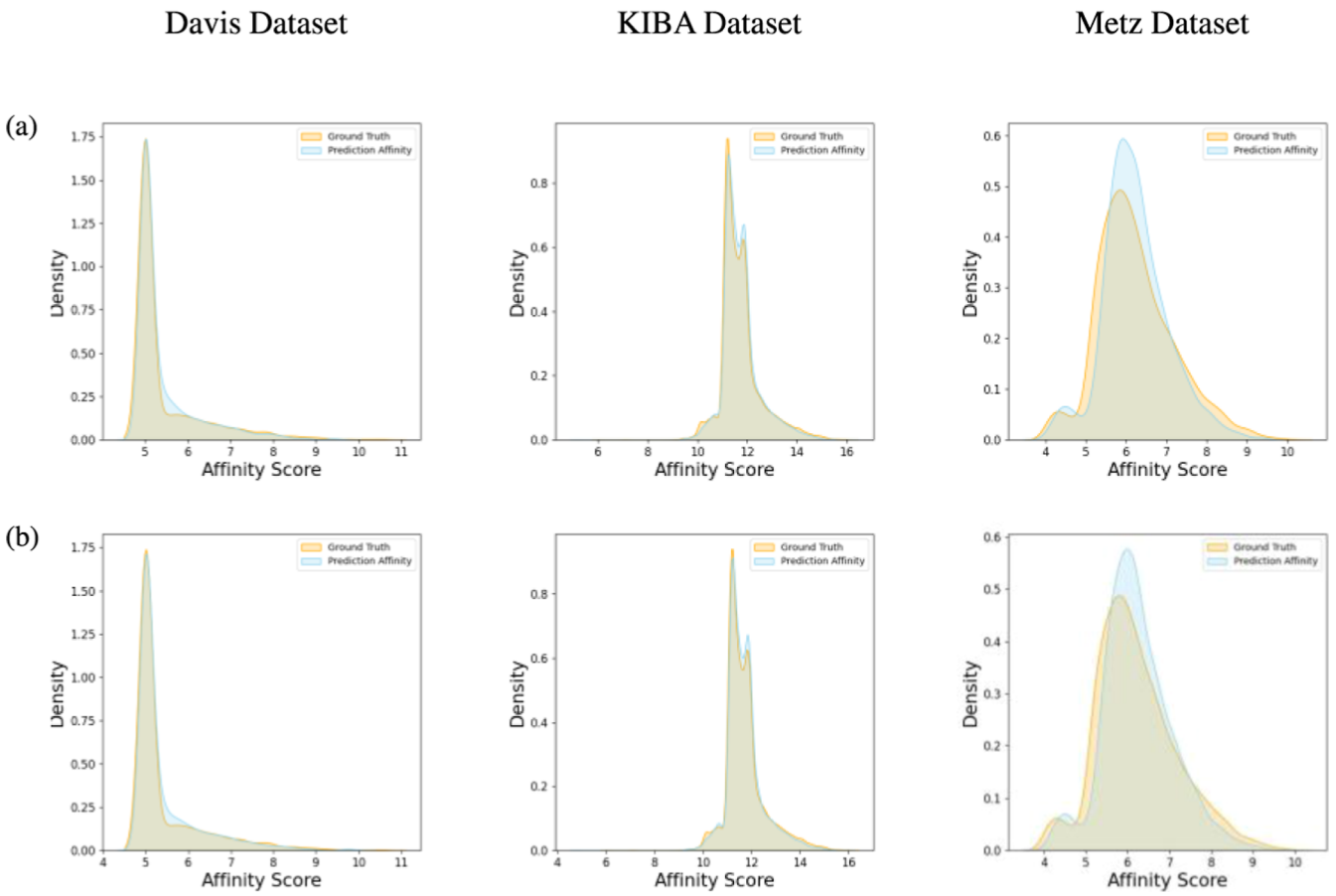


Figure 3. Visualization of the kernel density estimate plots of the predicted affinity values and ground truths for (a) MRBDTA and (b) BTDHDTA methods on the Davis, KIBA, and Metz data sets.

0.9528, and 0.8920 on the Davis, KIBA, and Metz data sets, respectively, while MRBDTA achieved overlap areas of 0.9005, 0.9412, and 0.8816. This demonstrates that the values predicted by the BTDHDTA model better reflect the true DTA.

4. CASE STUDY

To demonstrate the effectiveness of the BTDHDTA model in realistic scenarios, we employed drug repurposing, a novel strategy for drug discovery. Due to the global spread of the new coronavirus (SARS-CoV-2) in recent years, finding effective treatments has become an urgent task. Drug repurposing, which involves using computational methods to apply FDA-approved drugs or compounds that have passed clinical trials to new diseases or indications, is considered a potential approach for discovering treatments for the new coronavirus.[45,46] In this case study, our trained model was used to predict the binding affinity scores between 3137 FDA-approved drugs and six SARS-CoV-2 replication-related proteins.

According to previous studies,[47,48] we extracted the FASTA sequences of six SARS-CoV-2 replication-related proteins from the National Center for Biotechnology Information (NCBI) database. These proteins contain 3C-like proteinase, RNA-dependent RNA polymerase, helicase, 3′−5′ exonuclease, endoRNase, and 2′-*O*-ribose methyltransferase. Based on the literature,[39] we obtained the SMILES sequences for 3137 FDA-approved drugs, including 58 antiviral drugs. We then input these SMILES and protein sequences into the BTDHDTA model, trained on the KIBA data set, to predict the KIBA scores as binding affinities. The 3137 drugs were ranked based on their predicted binding affinities. To evaluate our model, we compared the ranking predictions to those from the MRBDTA model, ensuring that the hyperparameters and some internal settings of the MRBDTA model were consistent with ours. Our observations show that BTDHDTA can predict more antiviral drugs and rank them higher (see Table 9). Specifically, when using the KIBA scores as the binding affinity, BTDHDTA identified 22 antiviral drugs among the top 200, compared to 18 predicted by MRBDTA. For SARS-CoV-2 replication-related proteins, except for the 3C-like proteinase, BTDHDTA predicted as many or more antiviral drugs than MRBDTA and ranked them higher. Overall, these results indicate that our BTDHDTA model performs better than that of MRBDTA in drug repurposing.

Our predicted results are theoretical outcomes for the drug repurposing task and require validation through specific biological experiments. Notably, among the binding affinity predictions for all six SARS-CoV-2-related proteins, Saquinavir and Indinavir were both predicted as high-affinity drugs, ranking very high. This suggests that these two drugs could potentially be effective treatments against the new coronavirus. Furthermore, Remdesivir, which was predicted by BTDHDTA, has already been approved by the FDA and is the first drug authorized for treating the new coronavirus. Therefore, we hope that our work can provide some insights for new drug discovery and assist in the treatment of patients infected with the new coronavirus.

https://doi.org/10.1021/acsomega.4c08048
ACS Omega 2025, 10, 2020−2032

Table 9. For the Six SARS-CoV-2 Replication-Related Proteins, the Antiviral Drugs in Top 200 with Superior Binding Affinity Predicted by BTDHDTA and MRBDTA Based on KIBA Scores

proteins in SARS-CoV-2	BTDHDTA			MRBDTA		
	antiviral drug	KIBA score	rank of 3137 drugs	antiviral drug	KIBA score	rank of 3137 drugs
3C-like proteinase	saquinavir	13.4380	20	Oseltamivir acid	12.4670	28
	indinavir	12.2620	160	Etravirine (TMC125)	12.3957	33
				Nelfinavir	11.7873	132
				VX-222 (VCH-222)	11.6982	190
RNA-dependent RNA polymerase	saquinavir	12.9645	31	Oseltamivir acid	12.1391	134
	indinavir	12.2352	104	Amprenavir (agenerase)	12.0528	198
	Zanamivir	12.0702	132			
	Remdesivir	11.9693	177			
helicase	saquinavir	13.2245	7	Oseltamivir acid	12.2239	37
	indinavir	12.1834	69	Etravirine (TMC125)	11.8820	118
	Telaprevir (VX-950)	12.0382	94	Nelfinavir Mesylate	11.7929	162
	Nevirapine	11.9108	121			
	MK-5172	11.7260	185			
3′-to-5′ exonuclease	saquinavir	13.0063	11	Etravirine (TMC125)	12.4863	63
	indinavir	12.3474	38	Telaprevir (VX-950)	12.3502	99
	Nevirapine	11.8380	134	Oseltamivir acid	12.2393	155
endoRNase	saquinavir	12.6577	18	Oseltamivir acid	12.4711	47
	indinavir	11.9128	103	Nelfinavir Mesylate	12.0187	126
	Telaprevir (VX-950)	11.7849	142	Penciclovir	11.9203	177
	Lopinavir	11.6969	180	Telaprevir (VX-950)	11.9165	180
2′-*O*-ribose methyltransferase	saquinavir	12.7893	15	Etravirine (TMC125)	12.3507	54
	indinavir	12.1495	53	Oseltamivir acid	12.3374	57
	Nevirapine	11.8254	104			
	Telaprevir (VX-950)	11.6989	147			

5. CONCLUSIONS AND FUTURE WORKS

In this study, we investigate the problem of DTA prediction based on sequence representations and propose our DTA prediction model, BTDHDTA. Our model employs BiGRU after token embedding and position embedding to capture temporal relationships of the sequence representation. Based on the transformer encoder and dilated convolution, we construct a Trans block and a Dilated-CNN block, respectively. These two blocks are used to obtain rich feature representations of drugs and proteins. To effectively fuse the drug and protein representations that contain both local and global features, we employ a fusion module combining CNN with the Highway connection. The fusion module can regulate the flow of fused information and learn the relationships between drugs and proteins.

We test our model's performance on three public data sets and experimentally validated the effectiveness of each module. The results reveal the following key insights: (i) Capturing the temporal information on drug and target sequences is beneficial for the feature extraction module to extract key features from the sequences. (ii) Combining global features and multiscale local features, the model can learn more comprehensive and rich patterns from different levels of information. (iii) Modeling efficient feature fusion methods can learn interactions between drugs and proteins effectively, further improving the performance of DTA prediction. The BTDHDTA algorithm designed in this work achieves the best performance in most metrics and can accurately predict DTA, contributing to the sequence-based DTA prediction. Moreover, we apply the trained model to predict the binding affinities between six SARS-CoV-2 replication-related proteins and 3137 FDA-approved drugs. Two highly potential antiviral drugs are identified, providing valuable insights into the development of treatments against the novel coronavirus.

Although our BTDHDTA model has achieved good results in sequence-based DTA prediction, there is still much room for improvement. First, the BTDHDTA model currently focuses only on sequence information, and it is unclear whether it is still valid for structural information. Second, the model has not been applied to other prediction fields. Nowadays, studies targeting the relationship between ncRNAs and drug targets are getting more attention, and we hope to extend our model to the field of small molecule-miRNA association prediction (MMA) in future studies, a direction that has been strongly supported by many existing works.[50,51] The future direction of DTA prediction focuses on two aspects: data imbalance issue and model interpretability. The data imbalance issue often leads to poor DTA prediction performance, while the "black box" nature of deep learning makes the internal decision-making process of the model difficult to understand, which affects the acceptance of the results. Therefore, addressing the issue of data imbalance and enhancing model interpretability are also the focus of our future work.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acsomega.4c08048.

All materials and experimental results of case study, including FASTA sequence information for 6 proteins extracted from the NCBI database (PDF)

SMILES sequence information for 3137 FDA-approved drugs; predicted binding affinity scores (KIBA scores) of the BTDHDTA model and the baseline MRBDTA model; and all ranked prediction results (XLSX)

https://doi.org/10.1021/acsomega.4c08048
ACS Omega 2025, 10, 2020−2032

AUTHOR INFORMATION

Corresponding Author

Yuni Zeng − *School of Computer Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China;* orcid.org/0009-0007-1613-810X; Email: yunizeng@zstu.edu.cn

Authors

Zepeng Li − *School of Computer Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China*

Mingfeng Jiang − *School of Computer Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China*

Bo Wei − *School of Computer Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China; Longgang Research Institute, Zhejiang Sci-Tech University, Longgang 325000 Zhejiang, China*

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.4c08048

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This work was supported in part by the National Key Research and Development Program of China (2023YFF1205000 and 2023YFE0205600), the National Natural Science Foundation of China (62272415 and 62302456), the Key Research and Development Program of Zhejiang Province (2023C01041), the Key Research and Development Program of Ningxia Province (2023BEG02065), and the Zhejiang Provincial Natural Science Foundation of China (LQ23F020022).

REFERENCES

(1) Hopkins, A. L. Predicting promiscuity. *Nature* **2009**, *462* (7270), 167−168.

(2) Hughes, J. P.; Rees, S.; Kalindjian, S. B.; Philpott, K. L. Principles of early drug discovery. *Br. J. Pharmacol.* **2011**, *162* (6), 1239−1249.

(3) Chen, X.; Yan, C. C.; Zhang, X.; Zhang, X.; Dai, F.; Yin, J.; Zhang, Y. Drug−target interaction prediction: databases, web servers and computational models. *Briefings Bioinf.* **2016**, *17* (4), 696−712.

(4) Tsubaki, M.; Tomii, K.; Sese, J. Compound−protein interaction prediction with end-to-end learning of neural networks for graphs and sequences. *Bioinformatics* **2019**, *35* (2), 309−318.

(5) Yang, X.; Niu, Z.; Liu, Y.; Song, B.; Lu, W.; Zeng, L.; Zeng, X. Modality-dta: multimodality fusion strategy for drug−target affinity prediction. *IEEE/ACM Trans. Comput. Biol. Bioinf.* **2023**, *20* (2), 1200−1210.

(6) Zhu, Z.; Zheng, X.; Qi, G.; Gong, Y.; Li, Y.; Mazur, N.; Cong, B.; Gao, X. Drug−target binding affinity prediction model based on multi-scale diffusion and interactive learning. *Expert Syst. Appl.* **2024**, *255*, 124647.

(7) Pahikkala, T.; Airola, A.; Pietilä, S.; Shakyawar, S.; Szwajda, A.; Tang, J.; Aittokallio, T. Toward more realistic drug−target interaction predictions. *Briefings Bioinf.* **2015**, *16* (2), 325−337.

(8) He, T.; Heidemeyer, M.; Ban, F.; Cherkasov, A.; Ester, M. Simboost: a read-across approach for predicting drug−target binding affinities using gradient boosting machines. *J. Cheminf.* **2017**, *9*, 24.

(9) Wang, H. Prediction of protein−ligand binding affinity via deep learning models. *Briefings Bioinf.* **2024**, *25* (2), bbae081.

(10) Wei, L.; Zhou, C.; Chen, H.; Song, J.; Su, R. Acpred-fl: a sequence-based predictor using effective feature representation to improve the prediction of anti-cancer peptides. *Bioinformatics* **2018**, *34* (23), 4007−4016.

(11) Li, J.; Pu, Y.; Tang, J.; Zou, Q.; Guo, F. Deepatt: a hybrid category attention neural network for identifying functional effects of dna sequences. *Briefings Bioinf.* **2021**, *22* (3), bbaa159.

(12) Öztürk, H.; Özgür, A.; Ozkirimli, E. Deepdta: deep drug−target binding affinity prediction. *Bioinformatics* **2018**, *34* (17), i821−i829.

(13) Zhao, Q.; Duan, G.; Yang, M.; Cheng, Z.; Li, Y.; Wang, J. Attentiondta: drug−target binding affinity prediction by sequence-based deep learning with attention mechanism. *IEEE/ACM Trans. Comput. Biol. Bioinf.* **2023**, *20* (2), 852−863.

(14) Öztürk, H.; Ozkirimli, E.; Özgür, A., Widedta: prediction of drug−target binding affinity. **2019**, arXiv:1902.04166. arXiv preprint.

(15) Zeng, Y.; Chen, X.; Peng, D.; Zhang, L.; Huang, H. Multi-scaled self-attention for drug−target interaction prediction based on multi-granularity representation. *BMC Bioinf.* **2022**, *23* (1), 314.

(16) Kalemati, M.; Zamani Emani, M.; Koohi, S. Bicomp-dta: drug−target binding affinity prediction through complementary biological-related and compression-based featurization approach. *PLoS Comput. Biol.* **2023**, *19* (3), No. e1011036.

(17) Nguyen, T.; Le, H.; Quinn, T. P.; Nguyen, T.; Le, T. D.; Venkatesh, S. Graphdta: predicting drug−target binding affinity with graph neural networks. *Bioinformatics* **2021**, *37* (8), 1140−1147.

(18) Jiang, M.; Li, Z.; Zhang, S.; Wang, S.; Wang, X.; Yuan, Q.; Wei, Z. Drug−target affinity prediction using graph neural network and contact maps. *RSC Adv.* **2020**, *10* (35), 20701−20712.

(19) Zhu, Z.; Yao, Z.; Zheng, X.; Qi, G.; Li, Y.; Mazur, N.; Gao, X.; Gong, Y.; Cong, B. Drug−target affinity prediction method based on multi-scale information interaction and graph optimization. *Comput. Biol. Med.* **2023**, *167*, 107621.

(20) Wang, P.; Zheng, S.; Jiang, Y.; Li, C.; Liu, J.; Wen, C.; Patronov, A.; Qian, D.; Chen, H.; Yang, Y. Structure-aware multimodal deep learning for drug−protein interaction prediction. *J. Chem. Inf. Model.* **2022**, *62* (5), 1308−1317.

(21) Zhang, L.; Wang, C.-C.; Chen, X. Predicting drug−target binding affinity through molecule representation block based on multi-head attention and skip connection. *Briefings Bioinf.* **2022**, *23* (6), bbac468.

(22) Abbasi, K.; Razzaghi, P.; Poso, A.; Amanlou, M.; Ghasemi, J. B.; Masoudi-Nejad, A. Deepcda: deep cross-domain compound−protein affinity prediction through lstm and convolutional neural networks. *Bioinformatics* **2020**, *36* (17), 4633−4642.

(23) Yuan, W.; Chen, G.; Chen, C. Y.-C. Fusiondta: attention-based feature polymerizer and knowledge distillation for drug−target binding affinity prediction. *Briefings Bioinf.* **2022**, *23* (1), bbab506.

(24) Wu, H.; Liu, J.; Jiang, T.; Zou, Q.; Qi, S.; Cui, Z.; Tiwari, P.; Ding, Y. Attentionmgt-dta: a multi-modal drug-target affinity prediction using graph transformer and attention mechanism. *Neural Network.* **2024**, *169*, 623−636.

(25) Zhu, Z.; Yao, Z.; Qi, G.; Mazur, N.; Yang, P.; Cong, B. Associative learning mechanism for drug-target interaction prediction. *CAAI Trans. Intell. Technol.* **2023**, *8* (4), 1558−1577.

(26) Zhang, L.; Wang, C.-C.; Zhang, Y.; Chen, X. Gpcndta: prediction of drug-target binding affinity through cross-attention networks augmented with graph features and pharmacophores. *Comput. Biol. Med.* **2023**, *166*, 107512.

(27) Zheng, S.; Li, Y.; Chen, S.; Xu, J.; Yang, Y. Predicting drug−protein interaction using quasi-visual question answering system. *Nat. Mach. Intell.* **2020**, *2* (2), 134−140.

(28) Fang, K.; Zhang, Y.; Du, S.; He, J. Colddta: utilizing data augmentation and attention-based feature fusion for drug−target binding affinity prediction. *Comput. Biol. Med.* **2023**, *164*, 107372.

(29) Zhu, Y.; Zhao, L.; Wen, N.; Wang, J.; Wang, C. Datadta: a multi-feature and dual-interaction aggregation framework for drug−target binding affinity prediction. *Bioinformatics* **2023**, *39* (9), btad560.

(30) Wang, J.; Hu, J.; Sun, H.; Xu, M.; Yu, Y.; Liu, Y.; Cheng, L. Mgpli: exploring multigranular representations for protein−ligand interaction prediction. *Bioinformatics* **2022**, *38* (21), 4859−4867.

(31) Davis, M. I.; Hunt, J. P.; Herrgard, S.; Ciceri, P.; Wodicka, L. M.; Pallares, G.; Hocker, M.; Treiber, D. K.; Zarrinkar, P. P.

Comprehensive analysis of kinase inhibitor selectivity. *Nat. Biotechnol.* **2011**, *29* (11), 1046−1051.
(32) Tang, J.; Szwajda, A.; Shakyawar, S.; Xu, T.; Hintsanen, P.; Wennerberg, K.; Aittokallio, T. Making sense of large-scale kinase inhibitor bioactivity data sets: a comparative and integrative analysis. *J. Chem. Inf. Model.* **2014**, *54* (3), 735−743.
(33) Metz, J. T.; Johnson, E. F.; Soni, N. B.; Merta, P. J.; Kifle, L.; Hajduk, P. J. Navigating the kinome. *Nat. Chem. Biol.* **2011**, *7* (4), 200−202.
(34) Gönen, M.; Heller, G. Concordance probability and discriminatory power in proportional hazards regression. *Biometrika* **2005**, *92* (4), 965−970.
(35) Roy, K.; Chakraborty, P.; Mitra, I.; Ojha, P. K.; Kar, S.; Das, R. N. Some case studies on application of "rm2" metrics for judging quality of quantitative structure−activity relationship predictions: emphasis on scaling of response data. *J. Comput. Chem.* **2013**, *34* (12), 1071−1082.
(36) Bian, J.; Zhang, X.; Zhang, X.; Xu, D.; Wang, G. Mcanet: shared-weight-based multiheadcrossattention network for drug−target interaction prediction. *Briefings Bioinf.* **2023**, *24* (2), bbad082.
(37) Zeng, X.; Chen, W.; Lei, B. Cat-dti: cross-attention and transformer network with domain adaptation for drug−target interaction prediction. *BMC Bioinf.* **2024**, *25* (1), 141.
(38) Aleb, N. A mutual attention model for drug target binding affinity prediction. *IEEE/ACM Trans. Comput. Biol. Bioinf.* **2022**, *19* (6), 3224−3232.
(39) Zeng, Y.; Chen, X.; Luo, Y.; Li, X.; Peng, D. Deep drug−target binding affinity prediction with multiple attention blocks. *Briefings Bioinf.* **2021**, *22* (5), bbab117.
(40) Pu, Y.; Li, J.; Tang, J.; Guo, F. Deepfusiondta: drug−target binding affinity prediction with information fusion and hybrid deep-learning ensemble model. *IEEE/ACM Trans. Comput. Biol. Bioinf.* **2022**, *19* (5), 2760−2769.
(41) Wang, J.; Wen, N.; Wang, C.; Zhao, L.; Cheng, L. Electra-dta: a new compound−protein binding affinity prediction model based on the contextualized sequence encoding. *J. Cheminf.* **2022**, *14* (1), 14.
(42) Pei, Q.; Wu, L.; Zhu, J.; Xia, Y.; Xie, S.; Qin, T.; Liu, H.; Liu, T.-Y.; Yan, R. Breaking the barriers of data scarcity in drug−target affinity prediction. *Briefings Bioinf.* **2023**, *24* (6), bbad386.
(43) Hua, Y.; Song, X.; Feng, Z.; Wu, X. Mfr-dta: a multi-functional and robust model for predicting drug−target binding affinity and region. *Bioinformatics* **2023**, *39* (2), btad056.
(44) Han, L.; Kang, L.; Guo, Q. Imagedta: A simple model for drug−target binding affinity prediction. *ACS Omega* **2024**, *9* (26), 28485−28493.
(45) Moriaud, F.; Richard, S. B.; Adcock, S. A.; Chanas-Martin, L.; Surgand, J.-S.; Ben Jelloul, M.; Delfaud, F. Identify drug repurposing candidates by mining the protein data bank. *Briefings Bioinf.* **2011**, *12* (4), 336−340.
(46) Dittmar, M.; Lee, J. S.; Whig, K.; Segrist, E.; Li, M.; Kamalia, B.; Castellana, L.; Ayyanathan, K.; Cardenas-Diaz, F. L.; Morrisey, E. E.; et al. Drug repurposing screens reveal cell-type-specific entry pathways and fda-approved drugs active against sars-cov-2. *Cell Rep.* **2021**, *35* (1), 108959.
(47) Abdel-Basset, M.; Hawash, H.; Elhoseny, M.; Chakrabortty, R. K.; Ryan, M. Deeph-dta: deep learning for predicting drug−target interactions: a case study of covid-19 drug repurposing. *IEEE Access* **2020**, *8*, 170433−170451.
(48) Beck, B. R.; Shin, B.; Choi, Y.; Park, S.; Kang, K. Predicting commercially available antiviral drugs that may act on the novel coronavirus (sars-cov-2) through a drug−target interaction deep learning model. *Comput. Struct. Biotechnol. J.* **2020**, *18*, 784−790.
(49) Tang, X.; Zhou, Y.; Yang, M.; Li, W. Tc-dta: predicting drug−target binding affinity with transformer and convolutional neural networks. *IEEE Trans. NanoBioscience* **2024**, *23*, 572−578.
(50) Chen, X.; Guan, N.-N.; Sun, Y.-Z.; Li, J.-Q.; Qu, J. Microrna-small molecule association identification: from experimental results to computational models. *Briefings Bioinf.* **2020**, *21* (1), 47−61.
(51) Zhou, Z.; Du, Z.; Jiang, X.; Zhuo, L.; Xu, Y.; Fu, X.; Liu, M.; Zou, Q. GAM-MDR: probing miRNA−drug resistance using a graph autoencoder based on random path masking. *Briefings Funct. Genomic* **2024**, *23* (4), 475−483.

https://doi.org/10.1021/acsomega.4c08048
ACS Omega 2025, 10, 2020−2032