Digital Discovery

ROYAL SOCIETY OF CHEMISTRY

PAPER

View Article Online
View Journal | View Issue

Cite this: *Digital Discovery*, 2025, **4**, 2994

MAARDTI: a multi-perspective attention aggregation model for the prediction of drug–target interactions

Xinke Zhan,[a] Tiantao Liu,[a] Changqing Yu,[b] Yu-An Huang,[c] Zhuhong You[c] and Shirley W. I. Siu *[a]

Accurate prediction of drug–target interactions (DTIs) is indispensable for discovering novel drugs and repositioning existing ones. Recently, numerous methods based on deep learning have made promising progress in DTI predictions. These methods often utilize a single attention mechanism, which limits their ability to capture the complex features of both drugs and proteins. As a result, feature representation can be incomplete, training can become more complex and prone to overfitting. These together can impair the generalizability of the model. To address these problems, we propose an end-to-end neural network drug–target interaction approach called Multi-perspective Attention AggRegating (MAARDTI). Here, a multi-perspective attention mechanism is introduced that combines channel attention and spatial attention to capture a more comprehensive feature representation. The dual-context refocusing module is used to enhance the attention representation capability and improve the generalizability of the model. Experiments show that our proposed model outperforms ten state-of-the-art methods in three public datasets, achieving AUC values of 0.8975, 0.9248, and 0.9330 in DrugBank, Davis and KIBA, respectively. In the cold-splitting test with novel targets, drugs, and their bindings, MAARDTI performs on par with some methods for cold drug predictions. It outperforms in predicting unseen targets and bindings, underscoring the effectiveness of the novel multi-perspective attention mechanism in challenging scenarios. Hence, MAARDTI has the potential to serve as an effective tool for rapid identification of novel DTIs in drug research.

Received 14th July 2025
Accepted 27th August 2025

DOI: 10.1039/d5dd00311c

rsc.li/digitaldiscovery

Introduction

The identification of drug–target interactions (DTIs) is an important area of research within the drug discovery pipeline, with significant implications for drug discovery, repositioning, rediscovery and drug reprofiling.[1,2] While high-throughput experiments remain the most reliable approach for determining the interaction of drugs toward their targets, they pose considerable challenges when scaling up for large-scale *in vitro* screening and *in vivo* validation, as they are prohibitively costly and time-consuming.[3,4] Despite extensive investments, the drug development process is still very lengthy, taking up an average of 10 to 15 years to complete. The drug approval rate by the Food and Drug Administration (FDA) has been notably low, with less than 10% of drug candidates successfully progressing from phase I to final approval.[5] With the rapid development of *in silico* approaches for predicting DTIs,[6–8] computational methods have attracted increasing attention. Nowadays, it is possible to make full use of various biological data of known DTIs to quickly narrow down the scope of drug screening and explore unknown functions of drugs and proteins. These advances greatly motivate the drug discovery research to develop more efficient and accurate DTI prediction methods.

In recent years, a number of machine learning-based methods have been developed to predict DTIs.[9–13] These methods typically use the amino acid sequences of proteins and the Simplified Molecular Input Line Entry System (SMILES) of drugs as input. For instance, Wang *et al.*[14] reported a computational method that extracts feature vectors from drug structures and protein sequences and employs rotation forest to predict DTIs. Li *et al.*[15] obtained PSSM features from protein amino acid sequences and substructure fingerprints from drug chemical structures. To avoid the curse of dimensionality, the authors used principal component analysis (PCA) to reduce feature dimensions, and finally applied the local binary pattern (LBP) to predict DTIs. The aforementioned methods have made significant progress in the field, particularly by narrowing the search space for potential drug–target candidates. However, they suffer from the limitation that manual feature selection is required prior to model construction. This process can introduce bias,

[a]*Centre for Artificial Intelligence Driven Drug Discovery, Faculty of Applied Sciences, Macao Polytechnic University, Macau SAR, China. E-mail: shirleysiu@mpu.edu.mo*
[b]*School of Information Engineering, Xijing University, Xi'an 710123, China*
[c]*School of Computer Science, Northwestern Polytechnical University, Xi'an 710072, China*

leading to poor model generalizability and insensitivity to noisy data.

Recently, deep learning[16,17] has become a research focus since it can learn the latent feature representation through backward propagation without the need to manually engineer features. Moreover, a number of deep learning frameworks have shown outstanding performance at affordable costs compared to classical machine learning methods. In this regard, Öztürk *et al.*[18] proposed a model called DeepDTA, which only considers one-dimensional sequence representation. It includes two convolutional neural network (CNN) blocks to extract features from the amino acid sequence of proteins and SMILES of drugs separately, and then feeds them into a fully connected network (FCN) to obtain the final prediction results. Lee *et al.*[19] developed DeepConv-DTI, which expands the diversity of proteins to include diverse protein lengths and various target protein classes. In these studies, extensive experiments were conducted to validate the effectiveness of CNN-based methods. The difference between DeepDTA and DeepConv-DTI is that DeepConv-DTI adopted the extended connectivity fingerprint (ECFP) algorithm to extract drug features. Interestingly, neither of the two methods takes into account interaction features of the known protein–drug pairs, but treats protein and drug separately. Meanwhile, Zheng *et al.*[20] proposed a novel end-to-end deep learning framework called DrugVQA to predict DTIs, which defines the prediction task as a classical visual question answering problem. The method employs dynamic convolutional neural network (DynCNN) and bidirectional long short-term memory (LSTM) to extract features from 2D pairwise distance maps and represent drugs using molecular linear notation. Zhu *et al.*[21] reported a drug–target affinity prediction model named RRGDTA. The framework enhances correlations between molecular substructures and contextual features through a multi-scale interaction module (MSI), captures local structural correlations *via* a rotary encoding module (ROE), and preserves critical interaction patterns using an association prediction module (APM) with intra-mask retention (IMR). Wei *et al.*[22] reported a model named LAM-DTI to address the sequence length discrepancy between drugs and targets, and a learnable association information matrix dynamically adjusts to capture DTI pair information, effectively identifying interactions.

In addition, due to the remarkable performance of the transformer network, attention-based and BERT-based methods[23,24] have also been successfully used for predicting DTIs.[25–29] Notably, Huang *et al.*[30] proposed a transformer-based model named MolTrans. In their research, a 2D binding map of proteins and drugs is used as input and the molecular representation features are extracted by the augmented transformer module. Zhu *et al.*[31] proposed TDGraphDTA, a transformer and diffusion-based model for drug–target affinity prediction. The framework integrates multi-scale information interaction to capture relationships between molecular substructures and employs a diffusion-based graph optimization module to enhance molecular graph representation and interpretability. Zhao *et al.*[32] proposed an end-to-end model named HyperAttentionDTI, which adopts the attention mechanism of feature matrices. This method utilizes the original features of proteins and drugs, but high spatio-temporal complexity and the CNN block receptive field limit its performance. Bian *et al.*[33] proposed a shared-weighted-based multi-head cross attention network, called MCANet, which uses the cross-attention mechanism to compute attended protein and drug features. Ouyang *et al.*[34] introduced a BERT-inspired model called Pre-trained Multi-view Molecular Representations (PMMRs), an innovative neural network approach that leverages pre-trained models to enhance the generalizability and accuracy of drug–target binding predictions. By integrating multi-view molecular representations, they attempted to address the challenges posed by limited and diverse training data. These studies have proposed increasingly complex models with a large number of parameters, which makes training of these models particularly difficult and prone to overfitting. Moreover, these attention-based methods primarily focus on the unitary inter-subsequence or inter-substructure but ignore the positional features of the subsequence or substructure.

In view of the problems above, we propose an end-to-end neural network approach called MAARDTI for predicting DTIs. The drug SMILES strings and protein amino acid sequences are utilized as the input of our model. Two independent CNN blocks are then used to extract the protein and drug embedding features. In addition, different from the single attention mechanism of earlier studies, the MAAR module is designed to generate an aggregating matrix which fuses channel attention and spatial attention for strengthening subspace representation. Meanwhile, the bi-contextual refocusing module is adopted, which fuses attention matrices to obtain a multi-perspective attention feature representation, for improving attention generalizability. Finally, the drug and protein feature vectors are concatenated and fed into the prediction block. We conducted extensive experiments on several widely used benchmark datasets and compared our results with state-of-the-art methods. The superior prediction results of our method demonstrate the effectiveness of MAARDTI in predicting DTIs. Overall, this work has the following novelties: (i) A novel computational method outperforming state-of-the-art methods in DTI prediction is proposed. (ii) A multi-perspective attention aggregating module is designed, which captures the channel and spatial attention features to strengthen the feature learning ability of the model. (iii) A bi-contextual refocusing module including the drug-contextual refocusing block and the protein-contextual refocusing block is designed to further improve multi-dimensional feature expression of attention. (iv) To the best of our knowledge, this is the first attempt in which the fusion attention weight matrix and aggregation matrix are utilized for protein and drug subspace feature representations. We demonstrate that effective multi-perspective attention fusion can improve DTI prediction.

Methods

In this study, we report a novel computational method called MAARDTI for predicting potential DTIs. This model mainly contains four parts: the embedding block, the MAAR module,

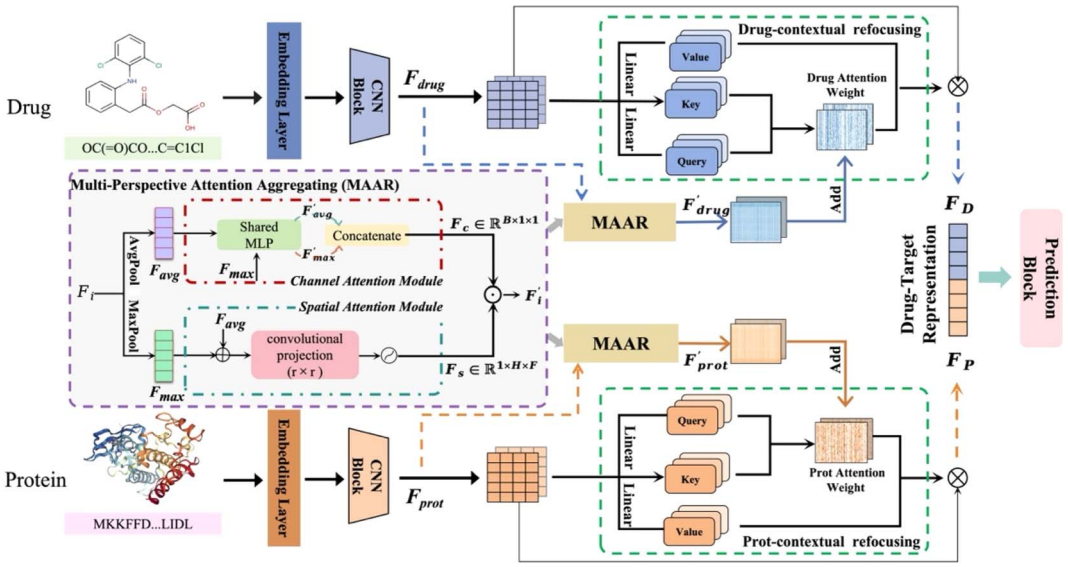

Fig. 1 The overall architecture of MAARDTI. Given a protein sequence and a drug SMILES string as an input, the prediction process consists of four main steps. (1) Embedding block: the categorical strings are passed through the embedding layer and the CNN block to extract low-dimensional feature representations. (2) MAAR module: the channel attention matrix and the spatial attention matrix are used to capture latent feature representations of substructures and subsequences. (3) Bi-contextual refocusing module: to learn protein and drug contextual information, the aggregation matrix and the attention weight matrix are fused to augment the attention ability representation. (4) Prediction block: the drug–target interaction is predicted based on the concatenated drug and protein features.

the bi-contextual refocusing module, and the prediction block. The architecture of the proposed model is depicted in Fig. 1. The core innovation of MAARDTI lies in its MAAR module, which integrates both channel and spatial attention mechanisms. This dual-attention approach allows the model to capture both the inter-channel relationships and spatial dependencies within the feature maps, thereby providing a more comprehensive representation of the interactions between drugs and proteins. Furthermore, the bi-contextual refocusing module enhances the feature fusion of drugs and proteins by incorporating contextual information from both sides, improving the model's ability to generalize and predict unseen interactions.

Drug–target embedding block

The raw data for our model are protein amino acid sequences and drug SMILES strings. Protein sequences are composed of 20 different amino acids and drug strings consist of 64 different characters. Prior to learning, these categorical values in the sequences and strings are encoded into numerical vector representations through the embedding layer by converting them into integer values. After embedding, the protein embedding matrix $P_{\text{embed}} \in \mathbb{R}^{L_{\text{prot}} \times D_{\text{p}}}$ and the drug embedding matrix $D_{\text{embed}} \in \mathbb{R}^{L_{\text{drug}} \times D_{\text{d}}}$ are obtained, where L_{prot} is the length of the protein sequence and L_{drug} is the length of the drug string, and D_{p} and D_{d} are the embedding dimension of the protein and drug, respectively. To enhance feature representation, we adopt two independent CNN blocks. Each CNN block contains three 1D-convolutional layers in which different sized filters are used for better capturing important local dependencies. The CNN block for the protein and the drug can be formulated as follows:

$$L_{\text{p}}^{(l+1)} = \sigma(\text{conv}(W_{\text{p}}^{(l)}, b_{\text{p}}^{(l)}, L_{\text{p}}^{(l)})) \quad (1)$$

$$L_{\text{d}}^{(l+1)} = \sigma(\text{conv}(W_{\text{d}}^{(l)}, b_{\text{d}}^{(l)}, L_{\text{d}}^{(l)})) \quad (2)$$

where $W_{\text{p}}^{(l)}$ and $W_{\text{d}}^{(l)}$ are the weight matrices and $b_{\text{p}}^{(l)}$ and $b_{\text{d}}^{(l)}$ are the biases in the l-th CNN layer. $L_{\text{p}}^{(l)}$ and $L_{\text{d}}^{(l)}$ are the hidden protein and drug representations in the l-th CNN layer. $\sigma(\cdot)$ denotes the ReLU activation function.

When the drug embedding matrix D_{embed} and the protein embedding matrix P_{embed} pass through the CNN block, the protein feature matrix $F_{\text{prot}} \in \mathbb{R}^{k_{\text{p}} \times m_{\text{p}}}$ and the drug feature matrix $F_{\text{drug}} \in \mathbb{R}^{k_{\text{d}} \times m_{\text{d}}}$ are generated in the latent feature space, where m_{p} and m_{d} are the dimensions of protein and drug feature vectors. Hence, these matrices contain semantic information and spatially associated information among the features.

Multi-perspective attention aggregating (MAAR)

We design a multi-perspective attention aggregating block that contains two attention sub-modules: the channel attention module and the spatial attention module. The following sections describe the specific details of each module.

Channel attention module. For exploiting the inter-channel relationship, the channel attention mechanism is utilized to enhance the selective attention features.[35] As different channels of feature maps may contain different important information, we constructed a channel attention module to capture meaningful sub-structures or sub-sequences. Given the input feature matrix $F_{\text{cnn}} \in \mathbb{R}^{B\times H\times D}$, the matrix is first fed into the max-pooling layer and the average-pooling layer to obtain average-pooled and max-pooled features, respectively.

$$F_{\text{avg}} = \text{avgpool}(F_{\text{cnn}}) \quad (3)$$

$$F_{\max} = \text{maxpool}(F_{\text{cnn}}) \tag{4}$$

where B is the number of channels and H and D are the height and weight of the matrix, respectively. Then, both F_{avg} and $F_{\max}$ are forwarded to a weight sharing network, which is composed of multilayer perceptron (MLP) with one hidden layer. In detail, the activation size of the hidden layer is set to $\mathbb{R}^{B/\rho\times1\times1}$, where ρ denotes the reduction ratio. After each descriptor is fed into the weight sharing network to obtain the channel attention map, $F_{\text{c}}\in\mathbb{R}^{B\times1\times1}$, element-wise summation operation is then applied to merge the output features. The definition of channel attention is computed as follows:

$$F_{\text{c}}(M) = \sigma(W_{\text{a}}(W_{\text{b}}(F^{\text{p}}_{\text{avg}})) + W_{\text{a}}(W_{\text{b}}(F^{\text{p}}_{\max}))) \tag{5}$$

where σ is the sigmoid function, and $W_{\text{a}}\in\mathbb{R}^{B/\rho\times B}$ and $W_{\text{a}}\in\mathbb{R}^{B\times B/\rho}$ are the weight matrices which are shared between both inputs F_{avg} and $F_{\max}$. The ReLU activation function is followed by W_{a}.

Spatial attention module. Spatial attention focuses on specific areas when processing data in computer vision to improve the performance and efficiency of the model.[36] In order to obtain the inter-spatial relationships, we build a spatial attention module, which focuses on identifying the significance of different areas within the subspace. More precisely, we obtain two-dimensional maps F_{avg} and $F_{\max}$ after the average-pooling and the max-pooling operations, and then they are concatenated to form one feature descriptor $F_{\text{s}}\in\mathbb{R}^{1\times H\times D}$. A convolution layer is employed to learn the 2D spatial attention matrix, which can be formulated as follows:

$$F_{\text{c}}(M) = \sigma(k^{n\times n}([F_{\text{avg}}; F_{\max}])) \tag{6}$$

where σ denotes the sigmoid function, $k^{n\times n}$ is the convolution operation and n is the kernel size.

Finally, the channel attention matrix $F_{\text{c}}(M)$ and the spatial attention matrix $F_{\text{s}}(M)$ are multiplied to obtain an aggregating attention matrix:

$$F_{\text{att}} = F_{\text{c}}(M) \times F_{\text{s}}(M) \tag{7}$$

where F_{att} is the final output feature representation from MAAR.

Bi-contextual refocusing module

In the attention mechanism,[37,38] three vectors—Query, Key, and Value—are crucial for effectively learning relationships within the data. To enhance our model, a transformer is adopted to map the input into different subspaces. Here, we perform a fusion operation between the aggregating attention map and the attention weight matrix to strengthen the attention of subspace by augmenting the substructure and subsequence feature representations.

After obtaining the aggregation attention matrix F_{att}, the input drug feature matrix F_{drug} and protein feature matrix F_{prot} are processed using the bi-contextual refocusing module. Taking drug feature as an example, the drug feature map is first projected to the Query matrix Q_{d}, the Key matrix K_{d} and the Value matrix V_{d}; then, we fuse the original attention matrix and the aggregation attention map. For each head, the combining attention result is calculated as follows:

$$\text{head}_{\text{d}} = \text{softmax}\left(\frac{Q_{\text{d}}{K_{\text{d}}}^{T}}{\sqrt{d_{\text{k}}}} + F_{\text{att}_{\text{d}}}\right)V_{\text{d}} \tag{8}$$

Afterwards, the drug output of the multi-head attention operation is computed as follows:

$$\text{Multihead}\ (Q, K, V) = \text{concat}_{d=1\ldots h}(\text{head}_{\text{d}})\,W^{\mathcal{O}} \tag{9}$$

where d represents the number of heads and $W^{\mathcal{O}}$ is the learnable matrix for feature mapping. We concatenate the outputs of all the heads and compute the final drug feature vector. Likewise, the protein operation is performed as follows:

$$\text{head}_{\text{p}} = \text{softmax}\left(\frac{Q_{\text{p}}{K_{\text{p}}}^{T}}{\sqrt{d_{\text{k}}}} + F_{\text{att}_{\text{p}}}\right)V_{\text{p}} \tag{10}$$

The protein output of the multi-head attention operation is as follows:

$$\text{Multihead}\ (Q, K, V) = \text{concat}_{d=1\ldots h}\left(\text{head}_{\text{p}}\right)W^{\mathcal{O}} \tag{11}$$

Hence, we concatenate the outputs of all the heads and obtain the final protein feature vector.

After applying the bi-contextual refocusing module, we obtain the drug augmentation matrix $D_{\text{aug}}\in\mathbb{R}^{k_{\text{d}}\times m_{\text{d}}}$ and the protein augmentation matrix $P_{\text{aug}}\in\mathbb{R}^{k_{\text{p}}\times m_{\text{p}}}$. The latent drug and protein feature matrices F_{D} and F_{P} are updated as follows:

$$\begin{aligned} F_{\text{D}} &= F_{\text{drug}}\cdot\alpha + D_{\text{aug}}\cdot\beta \\ F_{\text{P}} &= F_{\text{prot}}\cdot\alpha + P_{\text{aug}}\cdot\beta \end{aligned} \tag{12}$$

where both α and β are set to 0.5. This choice is based on the need to balance the contributions of the original and augmented features. The detailed results are shown in the Results section.

Prediction block

We then concatenate the drug and protein feature matrices F_{D} and F_{P}, and feed them into the multilayer fully connected networks (FCNs). The dropout layer is employed after the FCN layer to avoid overfitting during the training process. As a binary classification task, the output of the last layer is the probability of the interaction. Here, the PolyLoss function is used in this classification problem, which can efficiently mitigate the imbalance of the dataset. The loss function is computed as:

$$\mathcal{L}_{\text{poly}-1} = -\log(P) + \theta(1 - P) \tag{13}$$

where θ is the perturbation term with the value 1.

Experiments and results

Datasets

In this study, we train and evaluate the proposed MAAR model on three benchmark datasets: DrugBank,[39] Davis[40] and KIBA,[41] which have been widely used in previous studies.[42,43] The Davis

Table 1 Summary of the benchmark datasets for drug–target interactions

Datasets	Drug	Protein	Interaction	Positive	Negative
Davis	68	379	25 772	7320	18 452
KIBA	2068	225	116 350	22 154	94 196
DrugBank	6655	4294	35 022	17 511	17 511

and KIBA datasets consist of experimental assay values pK_d and pIC_{50} that measure binding affinities and the biological effect of drugs, respectively. It should be mentioned that in the KIBA and Davis datasets, the similarity among their drug molecules is low. Following the previous studies, the thresholds for true interactions are set as 5.0 for Davis and 12.1 for KIBA when constructing a binary classification dataset.[18] In detail, the Davis dataset provides binding affinities of 68 drugs toward 379 proteins, while KIBA provides binding affinities of 2068 drugs toward 225 proteins. Using DrugBank, we follow previous studies to construct a dataset containing drugs that are inorganic and small; those that cannot be recognized by RDKit[44] are manually discarded. As a result, there are 4294 proteins, 6655 drugs, and 17 511 validated positive DTIs in our DrugBank dataset. In order to obtain an equal number of negative DTIs, we randomly selected 17 511 unlabeled drug–target pairs to generate a complete dataset of 35 022 DTIs. The summary of the three benchmark datasets is shown in Table 1.

Implementation

Hyperparameters. Learning rate, batch size, and other hyperparameters are critical factors that can affect the prediction results. To train our model, the AdamW optimizer[45] is adopted with a learning rate of 1×10^{-4}. For the input data, the length of the drug SMILES strings is capped at 100 and the protein sequence length is 1000. The embedding layer is set to 64 dimensions for both proteins and drugs, which means that each amino acid or SMILES character has 64 dimensions. Due to the different lengths between drugs and proteins, the kernel sizes of the two parallel CNN blocks are different as well. The kernel sizes of the protein CNN blocks are 4, 8 and 12, whereas they are 4, 6, and 8 for the drug CNN blocks. The dropout rate is set to 0.1. Early stopping is employed to avoid the overfitting problem, and the patience parameter is set to 50. We train the proposed model using 300 epochs across all datasets. The detailed hyperparameters of the model are listed in Table S1 of SI.

Training setting. For all datasets, the ratio between the training and test sets is 4 to 1. The training set is further divided into five parts, with four parts used for training and one part for validation. After the optimal parameters are determined, we train a final model using the entire training set (80%) and evaluate the generalization performance of the model on the test set (20%). The MAARDTI model is implemented in PyTorch v1.12.1 (ref. 46) with CUDA version 11.3. All models are trained on a single NVIDIA GTX3090Ti GPU with 24 GB of memory.

Evaluation criteria. In this study, several commonly used evaluation metrics are adopted to better evaluate the effectiveness and robustness of our proposed method. We use the following performance metrics: accuracy, precision and recall. We also use the Area Under the receiver operating Characteristic curve (AUC) and the Area Under the Precision-Recall curve (AUPR) to measure the generalization performance of our proposed model.

Ensemble training. Five models are generated during the training process under five-fold cross validation. Since different training datasets result in models with different weights and sensitivities to drug–target pairs, combining these models into an ensemble model would provide higher prediction accuracy.

Results

Hyperparameter optimization

In the MAAR module, we combine channel attention and spatial attention features through a fusion operation. Five fusion strategies, namely addition, multiplication, maximum, average,

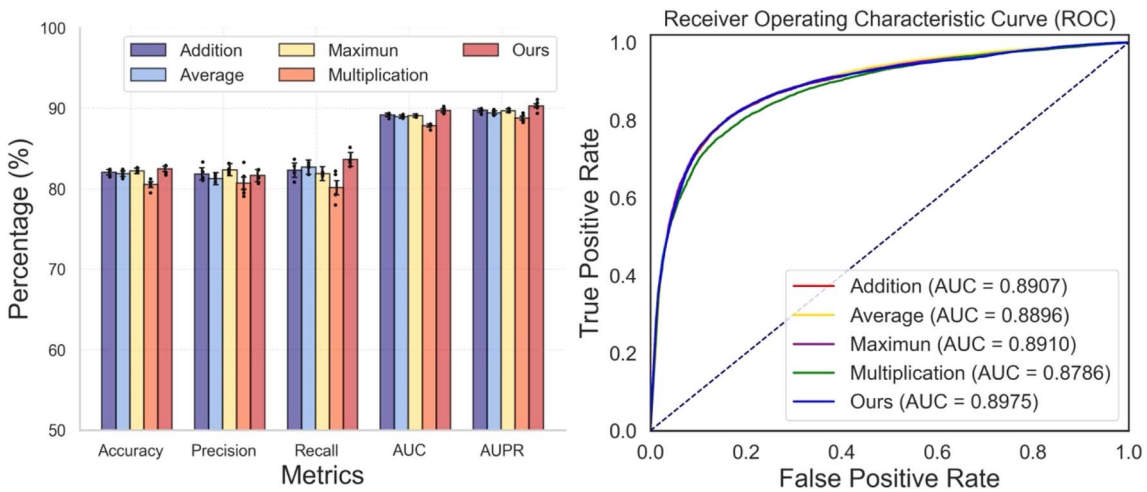

Fig. 2 Performance comparison and ROC curves of the five fusion strategies on the DrugBank dataset.

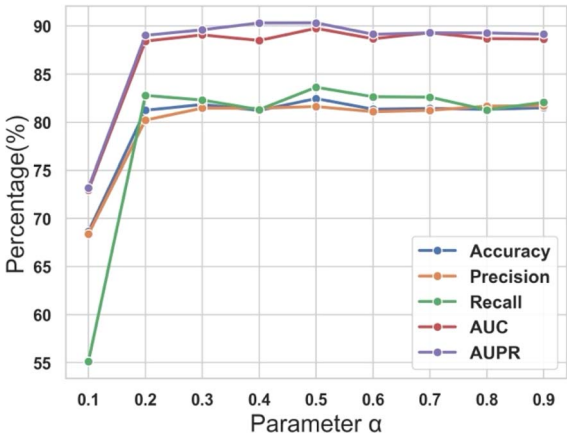

Fig. 3 The impact of fusion weights α and β on model performance.

and concatenation are compared empirically to decide for the best strategy. Using the DrugBank dataset and five-fold cross validation, the performance of the models is obtained. As shown in Fig. 2, the concatenation operation achieves the best overall performance with a high average accuracy of 0.8246, precision of 0.8163, recall of 0.8364, AUC of 0.8975, and AUPR of 0.9032. By concatenating features, all information of the two attention mechanisms is preserved and the model can learn how to best combine the features of these two attention mechanisms in subsequent layers, which increases the flexibility and adaptability of the model. Therefore, the concatenation operation is selected as the default fusion strategy for MARRDTI.

In addition, we experiment the fusion weights α and β used in the bi-contextual refocusing module. We set up nine groups of parameter experiments, with the α value ranging from 0.1 to 0.9, while β is set to $1 - \alpha$. As shown in Fig. 3, the model achieves peak performance across key metrics, including accuracy, AUC, and AUPR, when both α and β are set to 0.5. This validates our parameter selection, demonstrating that an equal weighting optimally balances the contributions of raw features and augmented features.

Cross-validation performance of MAARDTI

In this study, five-fold cross-validation is employed to mitigate the risk of overfitting during the training of the three benchmark datasets: DrugBank, Davis, and KIBA. Specifically, the entire dataset, excluding the test samples, is evenly divided into five parts, with four parts designated for training and one part designated for validation. The test dataset is then used for final predictions. The performance of the three datasets DrugBank, Davis and KIBA is shown in Tables 2–4.

Table 2 Five-fold cross-validation results of MAARDTI on the DrugBank dataset

Fold	Accuracy	Precision	Recall	AUC	AUPR
1	0.8287	0.8232	0.8358	0.9028	0.9113
2	0.8169	0.8066	0.8323	0.8933	0.8941
3	0.8286	0.8237	0.8346	0.8969	0.9026
4	0.8291	0.8143	0.8513	0.8999	0.9070
5	0.8195	0.8135	0.8277	0.8944	0.9008
Average	**0.8246**	**0.8163**	**0.8364**	**0.8975**	**0.9032**

Table 3 Five-fold cross-validation results of MAARDTI on the Davis dataset

Testing	Accuracy	Precision	Recall	AUC	AUPR
1	0.8766	0.7946	0.7637	0.9216	0.8442
2	0.8791	0.7855	0.7903	0.9284	0.8570
3	0.8740	0.7868	0.7637	0.9234	0.8494
4	0.8731	0.7876	0.7575	0.9280	0.8527
5	0.8702	0.7751	0.7650	0.9226	0.8445
Average	**0.8746**	**0.7859**	**0.7680**	**0.9248**	**0.8496**

Table 4 Five-fold cross-validation results of MAARDTI on the KIBA dataset

Testing	Accuracy	Precision	Recall	AUC	AUPR
1	0.8988	0.7269	0.7617	0.9334	0.8213
2	0.9005	0.7289	0.7713	0.9342	0.8196
3	0.8989	0.7233	0.7708	0.9334	0.8209
4	0.9002	0.7340	0.7559	0.9328	0.8160
5	0.9067	0.7350	0.7590	0.9305	0.8064
Average	**0.8998**	**0.7296**	**0.7637**	**0.9330**	**0.8168**

As shown in Table 2, our proposed method yields good performance on the DrugBank dataset, with high average values of 0.8246 for accuracy, 0.8163 for precision, 0.8364 for recall, 0.8975 for AUC, and 0.9032 for AUPR. When tested with the Davis imbalance dataset, it achieves high average values of 0.8746 for accuracy, 0.7859 for precision, 0.7680 for recall, 0.9248 for AUC, and 0.8496 for AUPR. Similarly, for the KIBA dataset, our method achieves high average values of 0.8998 for accuracy, 0.7296 for precision, 0.7637 for recall, 0.9330 for AUC, and 0.8168 for AUPR. Meanwhile, the ROC curves and PR curves of these three datasets are shown in Fig. 4 for visual comparison.

Comparative performance of MAARDTI against ten state-of-the-art methods

To better evaluate the performance of our proposed method, MAARDTI is compared against several competing DTI prediction methods. To ensure a fair comparison, these methods were installed locally and trained and tested on the same datasets using the five-fold cross-validation procedure as for our method. These baseline methods include two traditional machine learning models (Naïve Bayes and *K*-Nearest Neighbors), DeepDTA,[18] DeepConv-DTI,[19] MolTrans,[30] TransformerCPI,[28] HyperAttentionDTI,[32] MCANet,[33] MCANet-B,[33] RepConvDTI[47] and MGNDTI.[48] It is worth noting that the MCANet-B is an ensemble version of MCANet which combines five trained models into one integrated model, exhibiting improved performance. Using the same procedure, we train the MAARDITI's ensemble model, named MAARDTI-E for comparison. The results of the cross-validation tests and the ensemble

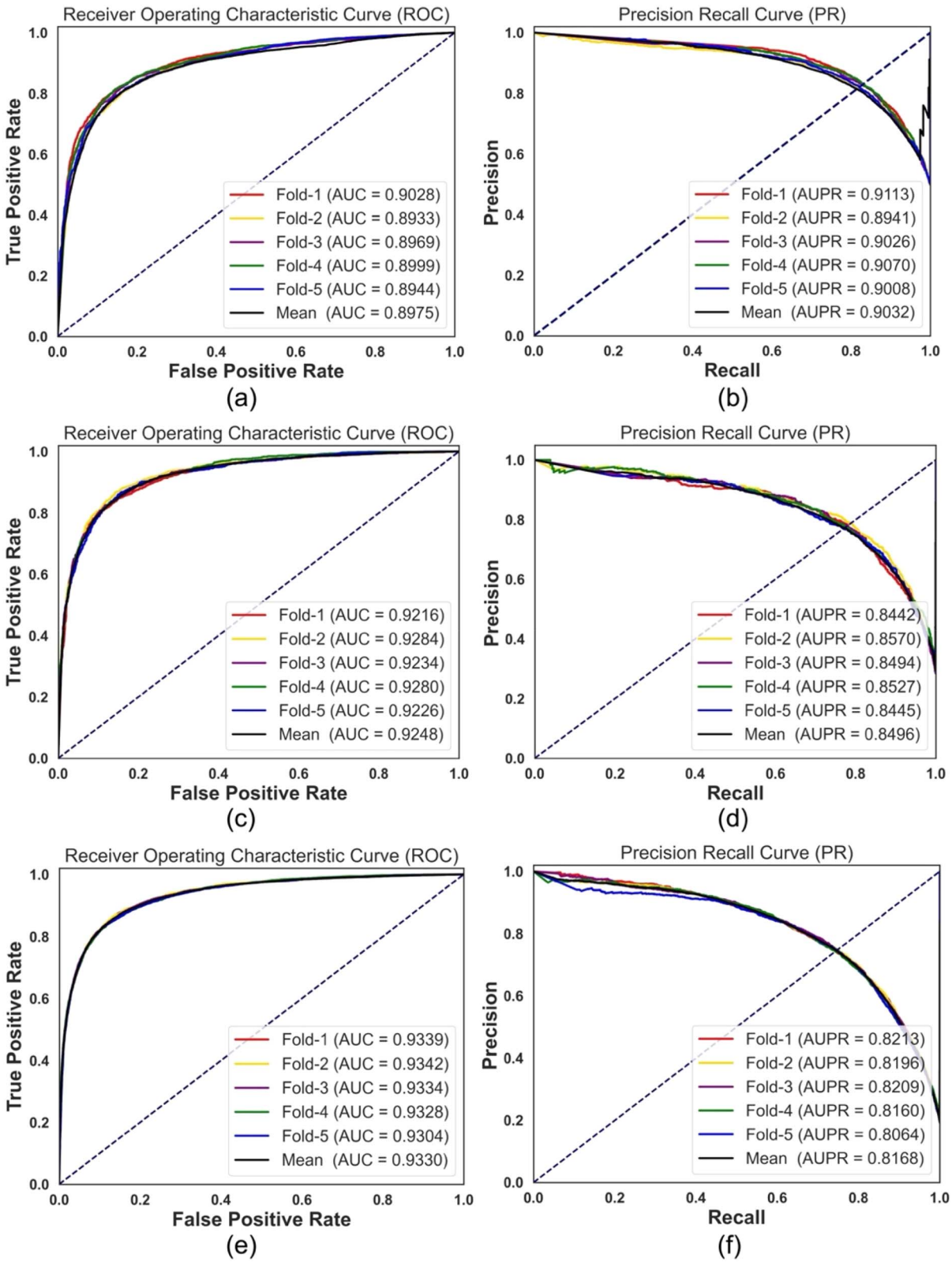

Fig. 4 The five-fold ROC curves and PR curves for the DrugBank (a and b), Davis (c and d) and KIBA (e and f) datasets.

results are shown in Tables 5–7, with the best method for a performance metric highlighted in bold and the second best italicized.

We train the ten baseline models with the DrugBank dataset. The experimental results in the five-fold cross-validation are summarized in Table 5. We can observe that our model achieves

Table 5 Benchmarking MAARDTI against ten methods using five-fold cross-validation on the DrugBank dataset

Models	Accuracy	Precision	Recall	AUC	AUPR
NB	0.5415	0.5468	0.5415	0.5417	0.6248
KNN	0.6736	0.6750	0.6736	0.6736	0.7526
DeepDTA	0.7772	0.7615	0.8052	0.8607	0.8679
DeepConv-DTI	0.7990	0.7942	0.7990	0.8683	0.8682
MolTrans	0.7621	0.7355	0.8264	0.8403	0.8407
TransformerCPI	0.7838	0.7722	0.8116	0.8565	0.8606
HyperAttentionDTI	0.8098	0.8026	0.8221	0.8900	0.8935
MCANet	*0.8180*	*0.8092*	*0.8309*	*0.8912*	*0.8963*
Rep-ConvDTI	0.7773	0.7620	0.7969	0.8590	0.8634
MGNDTI	0.8089	0.8037	0.8148	0.8816	0.8826
MAARDTI	**0.8246**	**0.8163**	**0.8364**	**0.8975**	**0.9032**
MCANet-B	0.8477	0.8461	0.8488	0.9089	0.9109
MAARDTI-E	**0.8524**	**0.8476**	**0.8581**	**0.9109**	**0.9203**

an improvement of 0.66%, 0.71%, 0.55%, 0.63% and 0.69% in accuracy, precision, recall, AUC and AUPR, respectively, over the best baseline model, MCANet. The results indicate that our model can predict DTIs more accurately.

We then train and assess our proposed model with the Davis dataset. Different from the DrugBank dataset, this is an imbalance dataset which is notoriously difficult to train and can lead to unrealistically high precision but low recall. As shown in Table 6, our model demonstrates enhanced performance on most metrics, achieving 0.55%, 0.42%, 0.26%, and 0.89% improvements in accuracy, recall, AUC and AUPR, respectively. It is noteworthy that although the precision of our model is slightly lower than that of DeepDTA, its recall is 9.3% higher, suggesting that our training strategy effectively addresses the challenge posed by dataset imbalance, resulting in enhanced generalization ability.

Furthermore, we train and assess our model with the KIBA dataset, which is also an imbalance dataset with nearly four times interaction pairs compared with the Davis dataset. Table 7 summarizes the comparative performance results. Our model achieves an improvement of 0.42% accuracy, 2.30% recall, 0.22% AUC and 0.42% AUPR over the best baseline MCANet. Meanwhile, the precision metric of our MAAR model is again lower than that of DeepConv-DTI but with a higher recall. This result further confirms that our training strategy is effective and that the resulting model is superior in predicting imbalance datasets. Finally, we examine the performance of the ensemble models. It is noteworthy that MAARDTI-E improves over its non-ensemble counterpart by 3.4, 2.3, and 1.6% on the DrugBank, Davis, and KIBA datasets, respectively, and it performs on par with MCANet-B.

Table 6 Benchmarking MAARDTI against ten methods using five-fold cross-validation on the Davis dataset

Models	Accuracy	Precision	Recall	AUC	AUPR
NB	0.6082	0.6429	0.6082	0.5641	0.4845
KNN	0.7240	0.6856	0.7240	0.5477	0.4703
DeepDTA	0.8568	0.7898	0.6776	0.9145	0.8337
DeepConv-DTI	0.8590	0.7761	0.7000	0.9163	0.8304
MolTrans	0.7847	0.6387	0.7121	0.8628	0.7299
TransformerCPI	0.8345	0.7400	0.6423	0.8863	0.7802
HyperAttentionDTI	0.8579	0.7428	0.7642	0.9142	0.8318
MCANet	*0.8691*	0.7750	*0.7602*	0.9222	0.8407
Rep-ConvDTI	0.8663	**0.7983**	0.7159	*0.9222*	*0.8439*
MGNDTI	0.8244	0.6553	0.7359	0.9093	0.8266
MAARDTI	**0.8746**	*0.7859*	**0.7680**	**0.9248**	**0.8496**
MCANet-B	0.8919	0.8260	**0.7848**	0.9441	0.8804
MAARDTI-E	**0.8946**	**0.8354**	0.7835	**0.9480**	**0.8956**

Table 7 Benchmarking MAARDTI against ten methods using five-fold cross-validation on the KIBA dataset

Models	Accuracy	Precision	Recall	AUC	AUPR
NB	0.6395	0.7231	0.6395	0.5570	0.3885
KNN	0.8206	*0.7911*	0.7206	0.5600	0.4667
DeepDTA	0.8931	0.7738	0.6324	0.9223	0.7935
DeepConv-DTI	0.7208	**0.7967**	0.6582	**0.9332**	**0.8212**
MolTrans	0.8891	0.7042	0.7353	0.9232	0.7949
TransformerCPI	0.8828	0.7087	0.6679	0.9070	0.7640
HyperAttentionDTI	0.8775	0.6730	0.7149	0.9161	0.7721
MCANet	0.8956	0.7277	0.7407	0.9308	0.8126
Rep-ConvDTI	0.8979	0.7763	0.6547	0.9255	0.8039
MGNDTI	0.8448	0.5611	**0.7766**	0.9227	0.8191
MAARDTI	**0.8998**	0.7296	*0.7637*	*0.9330*	*0.8168*
MCANet-B	0.9132	0.7852	0.7572	0.9488	0.8588
MAARDTI-E	**0.9143**	0.7788	**0.7771**	**0.9498**	**0.8609**

Ablation experiments

To better evaluate the effectiveness of the MAAR block and the attention fusion mechanism in our proposed model, we define three variants of the MAARDTI model:

- MAARDTI-OA: the MAAR block is removed and the output features of the CNN blocks are fed directly into the corresponding transformer modules.
- MAARDTI-PA: the drug MAAR block is removed and the protein MAAR block remains. For protein, the basic framework is the same as our proposed model. However, the output features of the CNN block for drugs are fed directly into the drug-contextual refocusing module.
- MAARDTI-DA: contrary to MAARDTI-PA, the protein MAAR block is removed and the drug MAAR block remains to predict drug–target interactions.

Table 8 Ablation experiments of MAARDTI using three variant models over five random runs

Dataset	Methods	Accuracy	AUC	AUPR	*t*-Value	*p*-Value
DrugBank	MAARDTI-OA	*0.8204*	0.8889	*0.8979*	3.485	<0.05
	MAARDTI-PA	0.8172	*0.8905*	0.8970	2.332	<0.05
	MAARDTI-DA	0.8131	0.8861	0.8948	8.829	<0.005
	MAARDTI	**0.8222**	**0.8948**	**0.9031**	—	—
Davis	MAARDTI-OA	0.8642	0.9160	0.8253	3.886	<0.005
	MAARDTI-PA	*0.8653*	0.9161	0.8237	3.774	<0.05
	MAARDTI-DA	0.8639	*0.9173*	*0.8318*	3.012	<0.05
	MAARDTI	**0.8703**	**0.9239**	**0.8454**	—	—
KIBA	MAARDTI-OA	*0.8912*	0.9240	*0.7960*	2.367	<0.05
	MAARDTI-PA	0.8899	*0.9249*	0.7955	2.897	<0.05
	MAARDTI-DA	0.8872	0.9225	0.7901	3.107	<0.05
	MAARDTI	**0.8982**	***0.9322***	***0.8117***	—	—

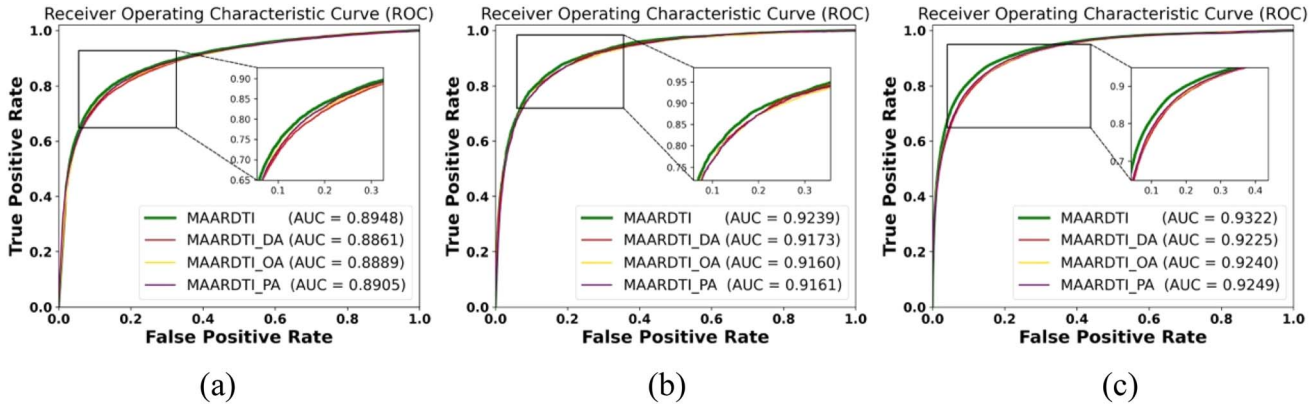

Fig. 5 The ROC curves of MAARDTI and the variant models for the ablation experiments on (a) DrugBank, (b) Davis, and (c) KIBA.

In addition, statistical tests, t-SNE, are conducted to evaluate the significance of the improvements reached by MAARDTI compared to each variant. Table 8 presents the prediction results with different variant models on the DrugBank, Davis and KIBA datasets from five random training models. Statistical tests are adopted to evaluate the significance of the improvement achieved by MAARDTI compared to each baseline model. Lower p values correspond to higher t values, with p values less than 0.05 indicating statistical significance. The results in Table 8 show that the MAARDTI model performs well in the DTI prediction with five random training models, outperforming its variants (MAARDTI-OA, MAARDTI-PA, and MAARDTI-DA). These results are statistically significant, further verifying the effectiveness and reliability of the MAARDTI model. The ROC curves of the three datasets are shown in Fig. 5 and we plot the attention heatmap and t-SNE feature distribution in Fig. 6. The heatmap shows the distribution of attention weights of different models on drug and protein features that brighter colors indicate higher attention weights, and the model pays more attention to these features. The heatmap of MAATDTI shows a more uniform and dispersed attention distribution. The t-SNE diagram shows the model's dimensionality reduction

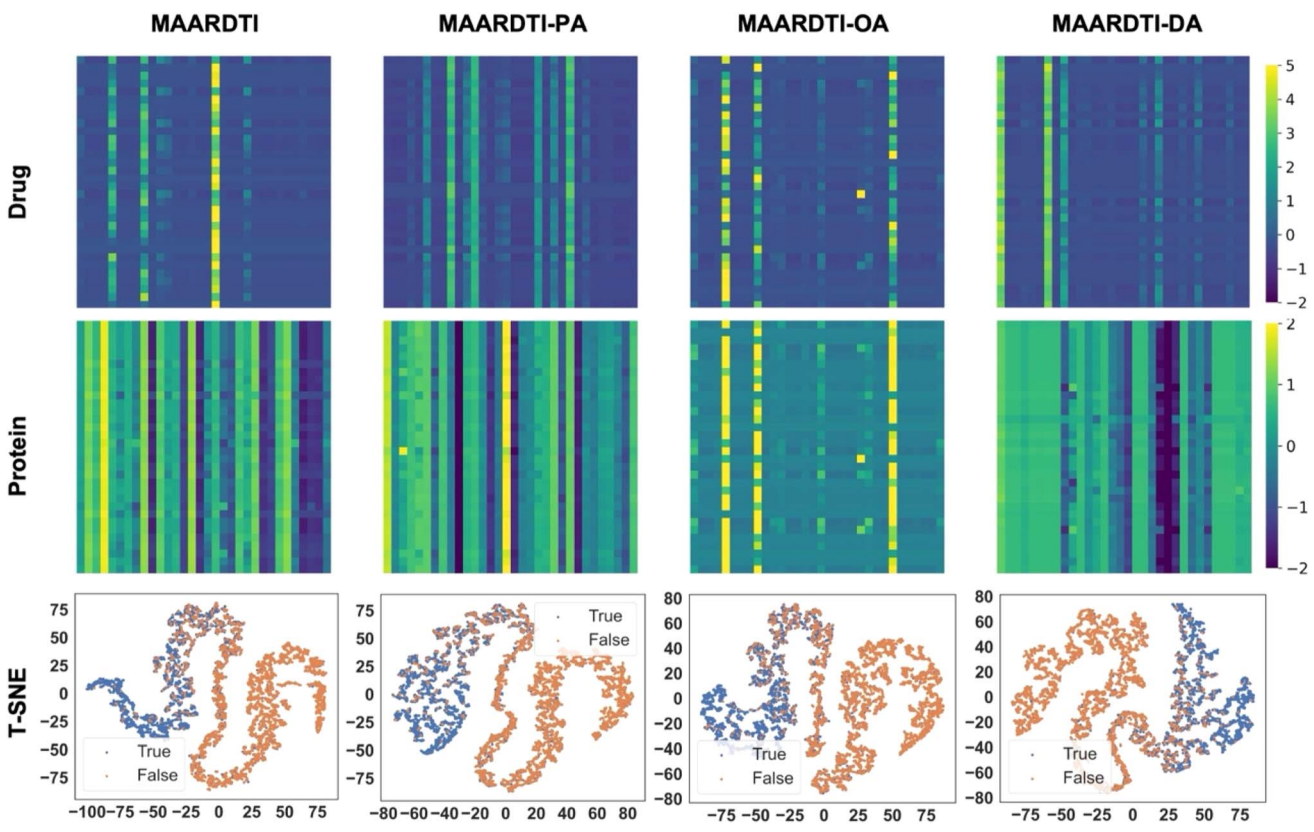

Fig. 6 Comparison of attention heatmaps and t-SNE visualization results of the four variant models of MAARDTI.

representation of DTI pair features in two-dimensional space. The blue points in the figure represent true interactions, and the red points represent non-interactions. The t-SNE graph of MAATDTI shows clearer clustering, and the boundaries between true interactions (blue dots) and non-interactions (red dots) are more obvious. This indicates that MAATDTI can better distinguish drug–protein interactions from non-interactions. We can conclude that MAATDTI performs better in drug–protein interaction prediction tasks. Its advantages in feature capture and classification boundaries enable it to more accurately predict drug–protein interactions. These advantages make MAATDTI a more reliable and effective tool that can be applied in drug discovery and biomedical research.

Case study 1 – DTI predictions for randomly selected untrained proteins and drugs

To validate the performance and generalization ability of our proposed model on real-world prediction tasks, we conducted several case studies following previous studies.[32,33] We randomly selected two drugs from the DrugBank dataset. The remaining data with the related drug–target pairs removed are set as the training set and the related pairs for testing. Two randomly selected drugs NADH (DB00157) and lisuride (DB00589) are adopted as testing data, and each drug contains 10 positive DTIs and 10 negative DTIs. Likewise, the same operation was performed on two randomly selected proteins DRD1 (P21728) and ADRA1D (P25100) as for the drugs. For drugs, NADH is a natural chemical that is involved in numerous enzymatic reactions. It is an approved nutraceutical supplement that is used in some dietary supplement products. On the other hand, lisuride (DB00589) is an ergot derivative that is an agonist for dopamine D2 receptors and some serotonin receptors. Table 9 lists the number of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) predictions and accuracy. The full results of predictions are given in the SI file. The accuracy of NADH (DB00157) and lisuride (DB00589) DTI predictions is 90% and 95%, respectively. For DRD1 (P21728) and ADRA1D (P25100), the accuracy of both reaches 95%, with one false prediction in each case. The above experimental results show that our proposed method has good prediction and generalization ability in predicting drug–target pairs.

Furthermore, to avoid biases caused by the coincidental selection of sequence or structurally similar proteins in the training set for the test set, we adopt MMseq2 (ref. 49) to cluster protein sequences in the DrugBank dataset. Then, we randomly remove a cluster of targets from the dataset, which contains five targets at least and uses it as a test set. The remaining data are used as the training set.

For protein clustering, we investigate the effects of clustering parameters on the performance of the model to simulate different scenarios as in real applications. The sequence identity (*seq-id*), coverage (*c*) and coverage mode (*cov-mode*) are three basic parameters in MMseq2. We build three cluster groups based on these parameters (Fig. 7):

(i) Strict group: this group has high similarity and coverage in each cluster and is suitable for assessing the risk of

Table 9 DTI prediction results of the two drugs NADH and lisuride and two proteins DRD1 and ADRA1D

Drug	True positive	True negative	False positive	False negative	Accuracy
DB00157-NADH	10	8	0	2	90%
DB00589-lisuride	10	9	0	1	95%
Protein	**True positive**	**True negative**	**False positive**	**False negative**	**Accuracy**
P21728-DRD1	10	9	0	1	95%
P25100-ADRA1D	10	9	0	1	95%

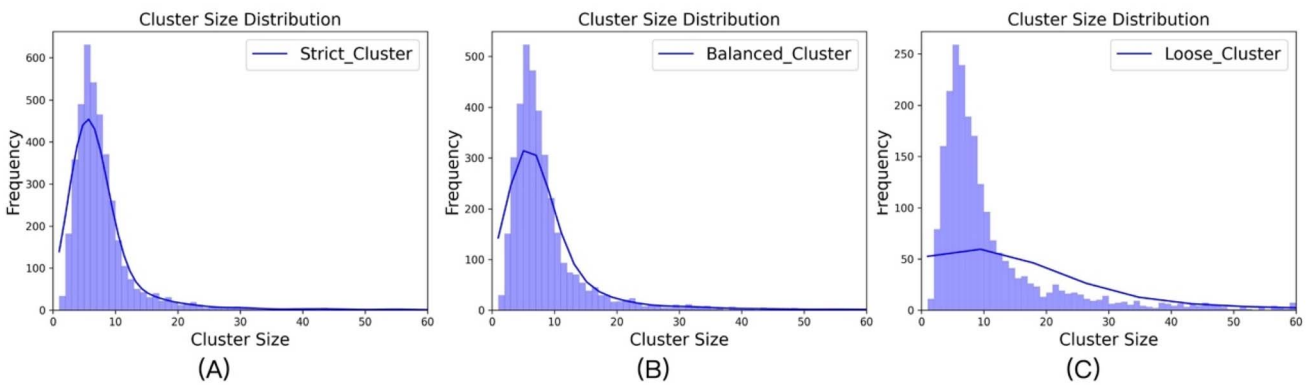

Fig. 7 The cluster size distribution of the (A) strict, (B) balanced, (C) loose groups of the DrugBank dataset.

Table 10 DTI prediction results of the three cluster groups based on different combinations of clustering parameters using MMSeq2

Cluster groups	Method	Accuracy	AUC	AUPR
Strict	MCANet	0.5625	0.8118	0.6977
	Ours	**0.7812**	**0.8275**	**0.7271**
Loose	MCANet	0.6562	0.7460	0.8081
	Ours	**0.8750**	**0.9667**	**0.9437**
Balanced	MCANet	0.8125	0.8213	0.6623
	Ours	**0.9375**	**0.8841**	**0.8722**

overfitting the model on closely related targets. The *seq-id* is set to 0.9, *c* is set to 0.9 and *cov-mode* is set to 1.

(ii) Balanced group: the number and size of clusters in this group are moderate. It reflects the real protein family distribution and is suitable for evaluating the universality of the model. The *seq-id* is set to 0.6, *c* is set to 0.7 and *cov-mode* is set to 2.

(iii) Loose group: this group has low similarity and low coverage, which can test the generalization ability of the model for cross-family targets. The *seq-id* is set to 0.3, *c* is set to 0.5 and *cov-mode* is set to 3.

Our model (MAARDTI) and MCANet (the second-best model in our benchmark) are tested in all clustering groups and the prediction results are shown in Table 10. Our model achieves a prediction accuracy of 78.12% for the strict group, 87.5% for the loose group, and 93.75% for the balanced group. Compared to its performance in the DrugBank benchmark (see Table 5) with an accuracy of 82.2%, the model has only a 5% lower performance in predicting unseen targets. However, based on the loose group result, the model has an improved accuracy of 6% in predicting distant targets. This indicates that our model has good generalizability. We focus more on AUC and AUPR since the number of positive and negative examples per protein is imbalanced, and accuracy could be misleading. We found that the AUC and AUPR values for the loose group were higher than those for the balanced and strict groups. We speculate that this is due to the following reasons: (i) the clustering structure of the loose group is looser, which allows the model to more flexibly capture the differences and connections between different samples when processing the data. (ii) The clustering structure of the loose group may be more consistent with the actual data distribution. (iii) The clustering structure of the loose group may allow the model to be exposed to a more diverse combination of samples during training. Compared to MCANet, our method is superior in accuracy by 8–40% in all clustering groups. These results show that our method is highly adaptive and robust in predicting test targets with different degrees of similarity to those in the training set.

Case study 2 – DTI predictions for cold drugs, cold targets, and cold bindings

Due to the inherent difficulties in collecting novel datasets for testing the generalizability of models, the cold splitting test approach has been widely used in drug research.[50,51] Following a previous study,[52] the cold target, cold drug, and cold binding were isolated from the BindingDB dataset. These cold drug–target pairs were excluded from both training and validation datasets. A cold drug refers to all interactions involving a drug that has not been previously encountered in the dataset, while a cold target encompasses all interactions with a target that is also novel to the model. In addition, the cold bindings comprise of interactions between cold targets and cold drugs. Overall, the number of cold targets, cold drugs and cold bindings is 136, 2127, and 114 respectively. Using this cold splitting test approach, we compare our model with MolTrans, HyperAttention, MCANet and DLM-DTI[52] for generalization performance. When we train DLM-DTI, we keep the hyperparameters consistent with the original operation. As can be seen in Fig. 8, our model MAARDTI is superior to four existing models in both cold binding and cold target predictions, achieving a 2% in AUC over the second-best method. For the cold drugs, while our model is on par with the other three models, DLM-DTI turns out to be the most accurate.

Case study 3 – DTI predictions on the olfactory receptor–molecule pair dataset

A key challenge in exploring the complexity of mammalian olfactory perception mechanisms is to understand how volatile organic compounds interact with olfactory receptors (ORs).[53] This understanding is crucial to revealing how odor molecules are recognized and processed by organisms, which in turn helps us understand how the olfactory system works and may provide new strategies for treating olfactory-related diseases. In order to address this challenge and develop models that can predict the ability of odor molecules to activate specific ORs, researchers need to rely on high-quality, broad-coverage datasets. The M2OR[54] dataset is a valuable resource created for this purpose and is a specially designed data resource for in-depth studies of the interactions between odor molecules and olfactory receptors. This dataset contains over 46 700 unique olfactory receptor (OR) and odor molecule pairs, which were carefully collated and analyzed from 31 scientific papers. It covers 11 different mammalian species, including 1237 unique OR sequences and 596 different molecules. To evaluate the ability of our model to predict for unseen proteins, we report the performance of the i.i.d. case, and followed the previous work[53] to predict in two scenarios, namely, random or cluster, *i.e.*, a randomly selected individual OR or a group of structurally similar ORs were put into the test set while removing their occurrences in the training set. The prediction results are presented in Table 11. Our model performs generally better in precision but has a low recall, which indicates that the model is more careful for positive sample prediction and could miss some true positive samples. As expected, predictions in the random scenarios are overall better than the cluster scenarios, in alignment with the model behavior reported by Matej *et al.* (performance values directly taken from this work).[53] In comparison, while our model has improved performance in terms of MCC by 19–83% in the cluster scenarios, the model by Hladiš *et al.* shows better generalization ability with enhanced MCC of 22–43% in the random scenarios. In the cluster group, the prediction results show that our model has better generalization ability at the

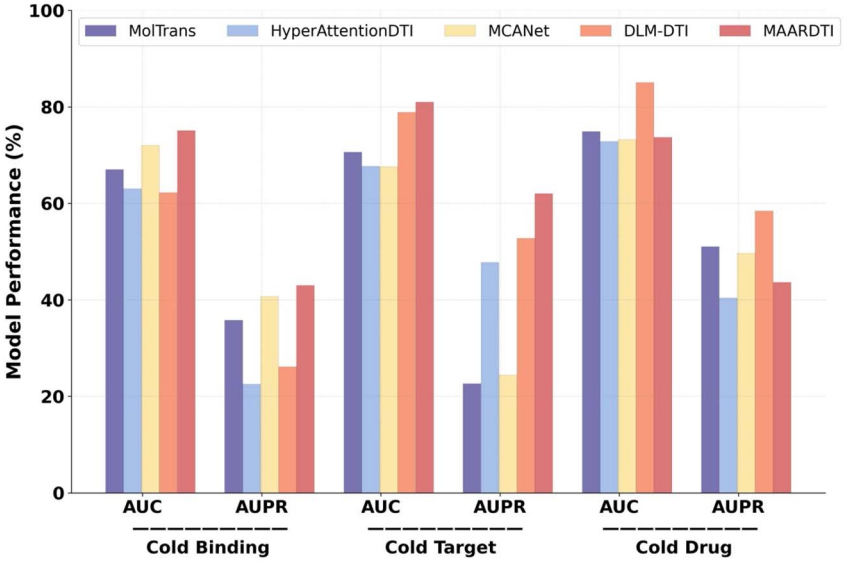

Fig. 8 The predictive performance of MAARDTI compared with four existing models in cold-splitting dataset prediction.

Table 11 Generalization test in two scenarios on the M2OR dataset

Split		Method	AveP	Precision	Recall	*F*-Score	MCC
	i.i.d.	Hladiš *et al.*[53]	0.780	0.689	0.698	0.693	**0.605**
		Ours	0.700	**0.700**	0.595	0.643	0.555
Cluster	Molecule	Hladiš *et al.*[53]	0.580	0.544	0.342	0.418	0.334
		Ours	0.423	0.795	0.242	0.371	**0.399**
	OR	Hladiš *et al.*[53]	0.558	0.535	0.132	0.203	0.088
		Ours	0.186	0.545	0.055	0.099	**0.147**
	OR-keep	Hladiš *et al.*[53]	0.625	0.576	0.095	0.161	0.091
		Ours	0.190	0.211	0.227	0.173	**0.167**
Random	Molecule	Hladiš *et al.*[53]	0.729	0.657	0.629	0.638	**0.533**
		Ours	0.445	0.633	0.344	0.446	0.409
	OR	Hladiš *et al.*[53]	0.684	0.636	0.491	0.552	**0.417**
		Ours	0.610	0.757	0.237	0.361	0.323
	OR-keep	Hladiš *et al.*[53]	0.710	0.670	0.470	0.548	**0.430**
		Ours	0.526	0.736	0.147	0.246	0.242

molecular level but has difficulty in handling a wider range of entities. In conclusion, the weak performance of both MAARDTI and Matej's model indicates that the M2OR dataset is highly challenging. Further efforts will focus on enhancing recall and generalization to unseen data.

Discussion

The identification of DTIs can facilitate the development of drug repositioning and accelerate the current drug discovery process. In recent years, a number of computational methods have been proposed to identify DTIs. These methods utilize different frameworks that mainly focus on how to fully extract drug and protein latent features such as semantic features, graph features, and meta-path features. In addition, transformer-based and BERT-based prediction frameworks have been rapidly developed. The ESM2 and BERT-related modules have shown their powerful feature extraction capabilities directly from protein sequences. Although most methods have achieved high performance on multiple public datasets, there is still much room for improvement to meet the requirements of real-world applications. In this paper, we propose a novel deep learning model, named MAARDTI, for predicting DTIs. To the best of our knowledge, this is the first attempt to fuse multi-perspective attention to enhance feature representations. This approach allows us to efficiently explore a large parameter space and fine-tune critical hyperparameters, resulting in a model that demonstrates superior performance across multiple benchmark datasets. The identification of optimal hyperparameters is crucial for achieving high performance in our proposed model. In this study, we employ grid search to systematically identify the best hyperparameters for our proposed model. Our optimization process begins with the definition of a comprehensive parameter space, which includes key hyperparameters such as learning rate, batch size, number of layers, dropout rates, and fusion weights. This step ensures

that we not only cover a wide range of possibilities but also focus on the most effective configurations. The optimized hyperparameters significantly contribute to the high performance of MAARDTI. For instance, the selected learning rate of 1×10^{-4} and fusion weights $\alpha = 0.5$ and $\beta = 0.5$ are found to be particularly effective in balancing model complexity and training efficiency. The choice of these hyperparameters also plays a crucial role in capturing local dependencies within the protein and drug sequences. We conducted a series of experiments with the balanced dataset DrugBank and two imbalanced datasets Davis and KIBA. For imbalanced datasets, we introduce the PolyLoss function, which focuses the model more on difficult samples and thus can improve the classification performance of minority class samples. The comparative experiments using extensive test data demonstrate that our model outperforms the state-of-the-art methods. Our model shows outstanding results in three public datasets, achieving AUC values of 0.8975, 0.9248, and 0.9330 in DrugBank, Davis and KIBA, respectively. It is worth noting that the KNN algorithm obtained the second-best performance in the precision metric in the KIBA dataset. This could be attributed to several factors. Firstly, the KIBA dataset may possess distinct local patterns that align well with the KNN algorithm's strength in capturing local similarities. Secondly, the algorithm's non-parametric nature and simplicity allow it to avoid overfitting, especially when the dataset is of moderate size and the feature space is relatively simple. However, this also suggests potential limitations in the KIBA dataset, such as a lack of complex global patterns or insufficient sample diversity, which might not fully challenge more sophisticated models. Moreover, the results of the ablation experiments show that the MAAR module, which exploits the drug–protein relationships *via* channel and spatial attention, can improve the prediction performance. Compared with other existing multi-attention or dual-attention methods, MAARDTI is unique in its comprehensive consideration of channel and spatial attention, as well as deep fusion of drug and protein features, which gives it stronger generalization ability and prediction performance when dealing with complex DTI prediction problems.

Although our proposed model has improved prediction performance, it still has some limitations. These include: (i) our model can only predict whether a protein and a drug interact without gaining insights into the underlying interaction mechanism. (ii) The potential of the multi-perspective attention mechanism for interpretability requires further exploration. (iii) While our optimization strategy proved effective, there is always room for improvement. The integration of automated hyperparameter tuning tools could streamline the optimization process and potentially uncover even more optimal configurations. (iv) The current single feature framework could limit the predictive performance of the model. Other embedding methods could be explored in future work. For example, pretrained protein language models have been shown to strengthen sequence latent representations, which can be combined with other sequence semantic information to obtain a more comprehensive representation. For drugs, graph representations, drug fingerprints, and drug motif representations can increase the diversity of feature representations to further improve the reliability of prediction. In the future, we would also like to apply MAARDTI on AI-based virtual screening of therapeutic targets for hit identification and drug reproposing, followed by validation through wet-lab experiments.

Conclusion

In this paper, we report a novel computational model, called MAARDTI, for predicting DTIs. We combine multi-attention features with a multi-perspective attention aggregation module to improve subspace feature representation while the bi-contextual refocusing block is used to detect the latent interrelationship of drugs and proteins. The comparison with the state-of-the-art methods demonstrates that our method has significantly improved performance. These outstanding results show that our proposed model MAARDTI can better capture the protein and drug latent features. We hope that this approach can help researchers to speed up drug screening and improve the success rate of drug discovery.

Conflicts of interest

No competing interest is declared.

Data availability

The DrugBank dataset is available at: **https://doi.org/10.1093/nar/gkj067**. The Davis dataset is available at: **https://doi.org/10.1038/nbt.1990**. The KIBA dataset is available at: **https://doi.org/10.1021/ci400709d**. The M2OR dataset is available at: **https://doi.org/10.1093/nar/gkad886**. Codes used for the experiments in this paper are available on Github at **https://github.com/TorchZhan/MAARDTI** or Zenodo at **https://doi.org/10.5281/zenodo.16936305**.

Supplementary information: detailed prediction results from the MAARDTI model and provide representative case studies of the model's predictive performance on drug-target interaction tasks. See DOI: **https://doi.org/10.1039/d5dd00311c**.

Acknowledgements

This project was funded by the National Natural Science Foundation of China under grant numbers 62273284, 62072378 and 62002297 and by Macao Polytechnic University with grant number RP/FCA-06/2024. The funders had no role in study design, data collection, and interpretation, or the decision to submit the work for publication. X. Z. and T. L. are recipients of the Macau Polytechnic University (MPU) graduate scholarship and would like to acknowledge MPU and NSFC for funding support. This project is part of the thesis work of X. Z. with an internal reference number of s/c fca.55e2.5650.e.

References

1 Y. Luo, X. Zhao, J. Zhou, *et al.*, *Nat. Commun.*, 2017, **8**(1), 573.

2 S. Tan, R. Lu, D. Yao, *et al.*, *ACS Chem. Neurosci.*, 2023, **14**(3), 481–493.
3 T. Hinnerichs and R. Hoehndorf, *Bioinformatics*, 2021, **37**(24), 4835–4843.
4 A. C. Nascimento, R. B. Prudêncio and I. G. Costa, *BMC Bioinf.*, 2016, **17**(1), 1–16.
5 S. Pushpakom, F. Iorio, P. A. Eyers, *et al.*, *Nat. Rev. Drug Discovery*, 2019, **18**(1), 41–58.
6 Q. Yuan, J. Gao, D. Wu, *et al.*, *Bioinformatics*, 2016, **32**(12), i18–i27.
7 Y. Wang, Z. You, S. Yang, *et al.*, *BMC Med. Inf. Decis. Making*, 2020, **20**(suppl.), 1–9.
8 F. Wan, L. Hong, A. Xiao, *et al.*, *Bioinformatics*, 2019, **35**(1), 104–111.
9 Z. Cheng, C. Yan, F. Wu and J. Wang, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, 2021, **19**(4), 2208–2218.
10 R. Lu, J. Wang, P. Li, *et al.*, *Brief. Bioinform.*, 2023, **24**(3), bbad145.
11 Z. Zhao, W. Huang, J. Pan, *et al.*, *Sci. Program.*, 2021, **2021**(1), 1–10.
12 X. Su, P. Hu, H. Yi, Z. You and L. Hu, *IEEE J. Biomed. Health Inform.*, 2022, **27**(1), 562–572.
13 H. He, G. Chen and C. Y. C. Chen, *Bioinformatics*, 2023, **39**(6), btad355.
14 L. Wang, Z. You, X. Chen, *et al.*, *J. Comput. Biol.*, 2018, **25**(3), 361–373.
15 Z. Li, P. Han, Z. You, *et al.*, *Sci. Rep.*, 2017, **7**(1), 11174.
16 K. Huang, T. Fu, L. M. Glass, *et al.*, *Bioinformatics*, 2020, **36**(22–23), 5545–5547.
17 Q. Ye, C. Y. Hsieh, Z. Yang, *et al.*, *Nat. Commun.*, 2021, **12**(1), 6775.
18 H. Öztürk, A. Özgür and E. Ozkirimli, *Bioinformatics*, 2018, **34**(17), i821–i829.
19 I. Lee, J. Keum and H. Nam, *PLoS Comput. Biol.*, 2019, **15**(6), e1007129.
20 S. Zheng, Y. Li, S. Chen, J. Xu and Y. Yang, *Nat. Mach. Intell.*, 2020, **2**(2), 134–140.
21 Z. Zhu, Y. Ding, G. Qi, *et al.*, *Eng. Appl. Artif. Intell.*, 2025, **147**, 110239.
22 Z. Wei, Z. Wang and C. Tang, *J. Chem. Inf. Model.*, 2025, **65**(8), 3915–3927.
23 R. Das, S. Jana, A. Dey, *et al.*, *IEEE Trans. Artif. Intell.*, 2025, 1–14.
24 Z. Lu, G. Song, H. Zhu, *et al.*, *Nat. Commun.*, 2025, **16**(1), 2548.
25 Z. Tian, X. Peng, H. Fang, *et al.*, *Brief. Bioinform.*, 2022, **23**(6), bbac434.
26 S. Lin, G. Zhang, D. Q. Wei and Y. Xiong, *Comput. Biol. Med.*, **149**, 105984.
27 Z. Zhu, Z. Yao, G. Qi, *et al.*, *CAAI Trans. Intell. Technol.*, 2023, **8**(4), 1558–1577.
28 L. Chen, X. Tan, D. Wang, *et al.*, *Bioinformatics*, 2020, **36**(16), 4406–4414.
29 Z. Zhu, X. Zheng, G. Qi, *et al.*, *Expert Syst. Appl.*, 2024, **255**, 124647.
30 K. Huang, C. Xiao, L. M. Glass and J. Sun, *Bioinformatics*, 2021, **37**(6), 830–836.
31 Z. Zhu, Z. Yao, X. Zheng, *et al.*, *Comput. Biol. Med.*, 2023, **167**, 107621.
32 Q. Zhao, H. Zhao, K. Zheng and J. Wang, *Bioinformatics*, 2022, **38**(3), 655–662.
33 J. Bian, X. Zhang, X. Zhang, D. Xu and G. Wang, *Brief. Bioinform.*, 2023, **24**(2), bbad082.
34 X. Ouyang, Y. Feng, C. Cui, *et al.*, *Bioinformatics*, 2025, **41**(1), btaf002.
35 J. Hu, L. Shen, and G. Sun, *IEEE Conf. Comput. Vis. Pattern Recognit.,* 2018, 7132–7141.
36 M. Jaderberg, K. Simonyan and A. Zisserman, *Adv. Neural Inf. Process. Syst.*, 2015, **28**, 2017–2025.
37 I. Lauriola, A. Lavelli and F. Aiolli, *Neurocomputing*, 2022, **470**, 443–456.
38 D. Khurana, A. Koli, K. Khatter and S. Singh, *Multimed. Tool. Appl.*, 2023, **82**(3), 3713–3744.
39 D. S. Wishart, C. Knox, A. C. Guo, *et al.*, *Nucleic Acids Res.*, 2006, **34**(suppl. 1), D668–D672, retrieved August 2024.
40 M. I. Davis, J. P. Hunt, S. Herrgard, *et al.*, *Nat. Biotechnol.*, 2011, **29**(11), 1046–1051, retrieved August 2024.
41 J. Tang, A. Szwajda, S. Shakyawar, *et al.*, *J. Chem. Inf. Model.*, 2014, **54**(3), 735–743, retrieved August 2024.
42 T. Pahikkala, A. Airola, S. Pietilä, *et al.*, *Brief. Bioinform.*, 2015, **16**(2), 325–337.
43 T. He, M. Heidemeyer, F. Ban, *et al.*, *J. Cheminform.*, 2017, **9**(1), 1–14.
44 G. Landrum, *RDKit: a software suite for cheminformatics, computational chemistry, and predictive modeling*, 2013, vol. 8, p. 5281, **https://www.rdkit.org/RDKit_Overview.pdf**.
45 I. Loshchilov and F. Hutter, *arXiv*, 2017, preprint, arXiv:1711.05101, DOI: **10.48550/arXiv.1711.05101**.
46 A. Paszke, S. Gross, F. Massa, *et al.*, *Adv. Neural Inf. Process. Syst.*, 2019, 32.
47 M. Deng, J. Wang, Y. Zhao, *et al.*, *Sci. Rep.*, 2025, **15**(1), 2579.
48 L. Peng, X. Liu, M. Chen, *et al.*, *J. Chem. Inf. Model.*, 2024, **64**(16), 6684–6698.
49 M. Steinegger and J. Söding, *Nat. Biotechnol.*, 2017, **35**(11), 1026–1028.
50 A. Chatterjee, R. Walters, Z. Shafi, *et al.*, *arXiv*, 2021, preprint, arXiv:2112. 13168, DOI: **10.48550/arXiv.2112.13168**.
51 J. Wang, N. Wen, C. Wang, *et al.*, *J. Cheminform.*, 2022, **14**(1), 14.
52 J. Lee, D. W. Jun, I. Song and Y. Kim, *J. Cheminform.*, 2024, **16**(1), 14.
53 M. Hladiš, M. Lalis, S. Fiorucci, and J. Topin, *Int. Conf. Learn. Represent.*, 2023.
54 M. Lalis, M. Hladiš, S. A. Khalil, *et al.*, *Nucleic Acids Res.*, 2024, **52**(D1), D1370–D1379.