nature machine intelligence

Article https://doi.org/10.1038/s42256-022-00605-1

Interpretable bilinear attention network with domain adaptation improves drug–target prediction

Contents

1 Clustering-based pair split strategy

As mentioned in the main text, we separately cluster drug compounds and target proteins of the BindingDB and BioSNAP datasets for cross-domain performance evaluation. Specifically, we choose the single-linkage clustering, a bottom-up hierarchical clustering to ensure that the distances between samples in different clusters are always larger than a pre-defined distance, i.e., minimum distance threshold γ. This property can prevent clusters from being too close to help to generate the cross-domain scenario.

We use binarized ECFP4 feature to represent drug compounds, and integral PSC feature to represent target proteins. For accurately measuring the pairwise distance, we use the Jaccard distance and cosine distance on ECFP4 and PSC, respectively. We choose $\gamma = 0.5$ in both drug and protein clusterings since this choice can prevent over-large clusters and be ensure separate dissimilar samples. We obtained 2,780 clusters of drugs and 1,693 clusters of proteins for the BindingDB dataset, and 2,387 clusters of drugs and 1,978 clusters of proteins for the BioSNAP dataset. Table 1 shows the number of samples in the ten largest clusters of the clustering results. It shows that BindingDB has a more balanced cluster distribution than BioSNAP in drug clustering. In addition, the protein clustering result tends to generate many small clusters with only a few proteins in both datasets, indicating that the average similarity between proteins is lower than that between drugs. We randomly select 60% drug clusters and 60% protein clusters from clustering result, and regard all associated drug-target pairs with them as source domain data. The associated pairs in the remaining clusters are considered to be source domain data. We conduct five independent clustering-based pair splits with different random seeds for downstream model training and evaluation. Clustering-based pair split allows quantitatively constructing cross-domain tasks by considering the similarity between drugs or proteins.

Table 1. Size of the ten largest clusters in the BindingDB and BioSNAP datasets generated by the clustering-based pair split.

Dataset	Object	# 1	# 2	# 3	# 4	# 5	# 6	# 7	# 8	# 9	# 10
BindingDB	Drug	598	460	304	290	253	250	203	202	198	158
BioSNAP	Drug	294	267	75	68	36	35	28	26	24	24
BindingDB	Protein	17	15	15	12	10	10	10	9	9	8
BioSNAP	Protein	8	8	8	6	5	4	4	4	4	4

2 Dataset statistics, notations, and preprocessing steps

Table 2 shows the statistics of experimental datasets and Table 3 lists the notations used in this paper with descriptions. The BioSNAP and Human datasets were created by Huang et al. (2021)[1] and Liu et al. (2015)[2], respectively. For the BindingDB dataset, we created a low-bias version from the BindingDB database source[3] following the bias-reducing preprocessing steps in our earlier work[4]: i) We considered a drug-target pair to be positive only if its IC50 is less than 100 nM, and negative only if its IC50 was greater than 10,000 nM, giving a 100-fold difference to reduce class label noise. These IC50 thresholds were selected following earlier works[5,6]. ii) We removed all DTI pairs where the drugs only had one type of pairs (positive or negative) to improve drug-wise pair class balance and reduce hidden ligand bias that can lead to the correct predictions based only on drug features.

Table 2. Experimental dataset statistics

Dataset	# Drugs	# Proteins	# Interactions
BindingDB[4]	14,643	2,623	49,199
BioSNAP[1]	4,510	2,181	27,464
Human[2]	2,726	2,001	6,728

3 Hyperparameter setting and sensitivity analysis

Table 4 shows a list of model hyperparameters and their values used in experiment. As our model performance is not sensitive to hyperparameter setting, we use the same hyperparamters on all experimental datasets (BindingDB, BioSNAP and Human). Figure 1 illustrates the learning curves with the different choices of hyperparameters on the BindingDB validation set, including bilinear embedding size, learning rate and heads of attention. It shows that the performance differences are not large and typically converges between 30 and 40 epochs.

Table 3. Notations and descriptions

Notations	Description
$\mathbf{E}_p \in \mathbb{R}^{23 \times D_p}$	protein amino acid embedding matrix
$\mathbf{f} \in \mathbb{R}^{K/s}$	drug-target joint representation
$F(\cdot), G(\cdot), D(\cdot)$	feature extractor, decoder and domain discriminator in CDAN
$\mathbf{g} \in \mathbb{R}^2$	output interaction probability by softmax function
$\mathbf{H}_p^{(l)}, \mathbf{H}_d^{(l)}$	hidden representation for protein (drug) in l-th CNN (GCN) layer
$\mathbf{I} \in \mathbb{R}^{N \times M}$	pair-wise interaction matrix between drug and protein substructures
$\mathbf{M}_d \in \mathbb{R}^{\Theta_d \times 74}$	drug node feature matrix by its chemical properties
$p \in \mathbb{R}^1$	output interaction probability by Sigmoid function
$\mathcal{P}, \mathcal{G}$	protein amino acid sequence, drug 2D molecular graph
$\mathbf{q} \in \mathbb{R}^K$	weight vector for bilinear transformation
$\mathbf{U} \in \mathbb{R}^{D_d \times K}$	the weight matrix for encoded drug representation
$\mathbf{V} \in \mathbb{R}^{D_p \times K}$	the weight matrix for encoded protein representation
$\mathbf{W}_c, \mathbf{b}_c$	the weight matrix and bias for protein CNN encoder
$\mathbf{W}_g, \mathbf{b}_g$	the weight matrix and bias for drug GCN encoder
$\mathbf{W}_o, \mathbf{b}_o$	the weight matrix and bias for decoder
$\mathbf{X}_p \in \mathbb{R}^{\Theta_p \times D_p}$	latent protein matrix representation
$\mathbf{X}_d \in \mathbb{R}^{\Theta_d \times D_d}$	latent drug matrix representation

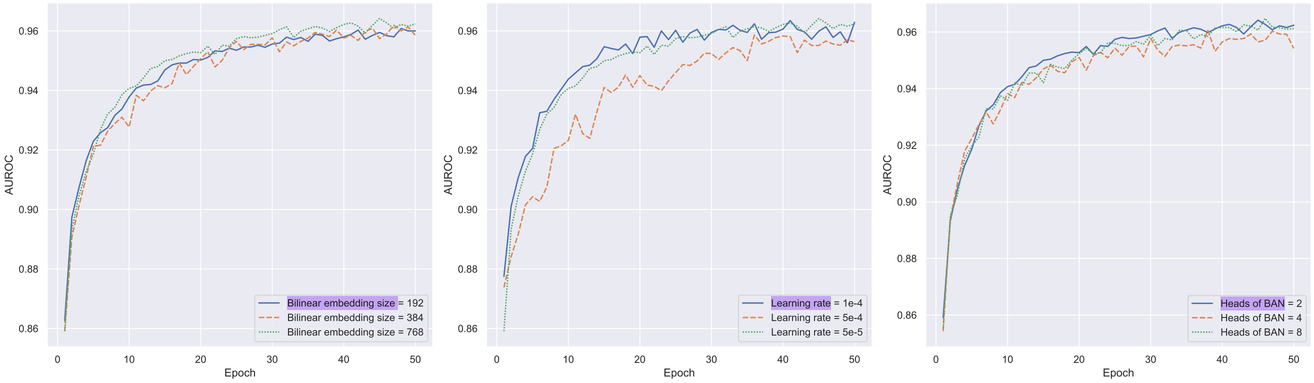

Figure 1. Learning curves with the different choices of hyperparameters on the BindingDB validation set.

4 Performance comparison across different protein families

We conduct experiments to study the performance of DrugBAN on different protein families. Following the previous studies[1,7], we select four major protein families: enzymes, G protein-coupled receptors (GPCRs), ion channels and nuclear hormone receptors (NHRs). We randomly retrieve one in-domain test set of BindingDB and BioSNAP respectively, and map their proteins to the four protein families using GtoPdb database (https://www.guidetopharmacology.org/targets.jsp). Table 5 presents the number of interactions for each protein family in the test sets. Figure 2 shows the performance (AUROC and AUPRC) varying only slightly given different protein families.

5 Performance comparison on unseen drugs/targets

To study how DrugBAN and other deep learning baselines perform on unseen drugs/targets, we conduct additional experiments on BindingDB and BioSNAP. For each dataset, we randomly select 20% drugs/target proteins. Then we evaluate predictive performance on all DTI pairs associated with these drugs/target proteins (70% as test set for evaluation and 30% as validation set for determining early stopping), and the rest pairs as training set for model optimization. Each unseen setting has five

Table 4. DrugBAN hyperparameter configuration

Module	Hyperparameter	Value
Optimizer	Learning rate	5e-5
Mini-batch	Batch size	64
Three-layer CNN protein encoder	Initial amino acid embedding	128
	Number of filters	[128, 128, 128]
	Kernel size	[3, 6, 9]
Three-layer GCN drug encoder	Initial atom embedding	128
	Hidden node dimensions	[128, 128, 128]
Bilinear interaction attention	Heads of bilinear attention	2
	Bilinear embedding size	768
	Sum pooling window size	3
Fully connected decoder	Number of hidden neurons	512
Discriminator	Number of hidden neurons	256

Table 5. Number of interactions for major protein families in the test sets.

Dataset	# Enzymes	# GPCRs	# Ion channels	# NHRs
BindingDB	5,277	472	440	144
BioSNAP	1,956	536	510	103

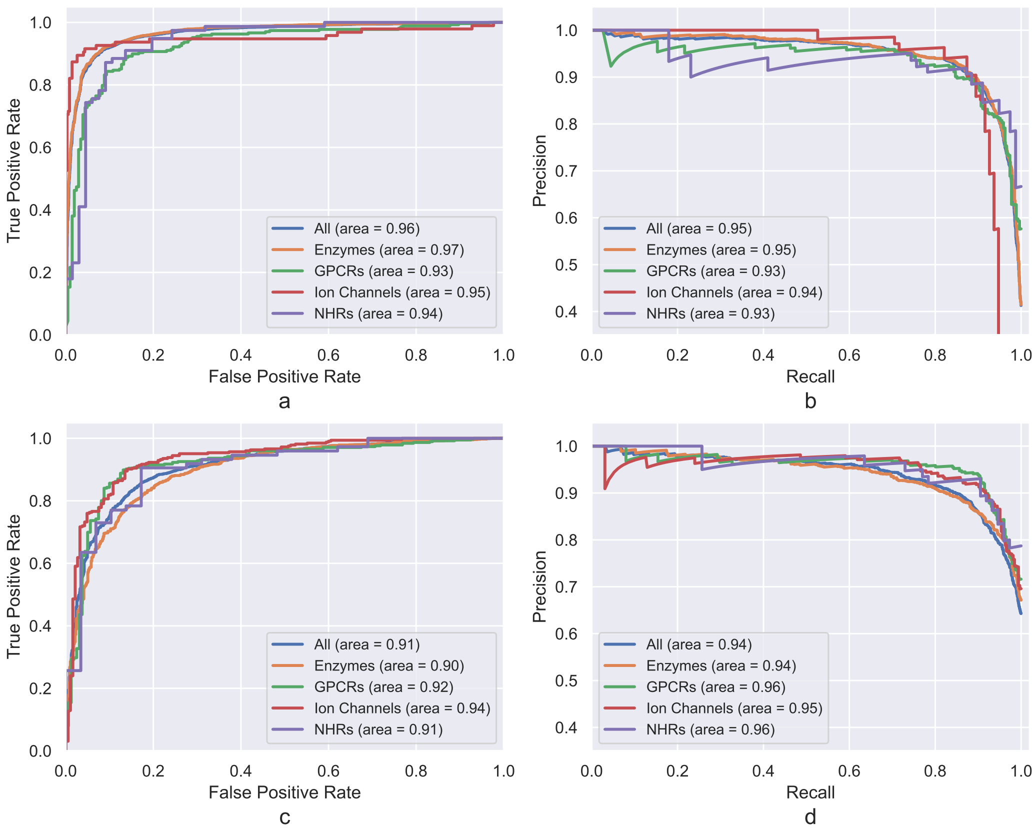

Figure 2. DrugBAN performance on different protein families. (a) AUROC curves on the BindingDB dataset. **(b)** AUPRC curves on the BindingDB dataset. **(c)** AUROC curves on the BioSNAP dataset. **(d)** AUPRC curves on the BioSNAP dataset.

Table 6. Performance (average AUROC over five random runs) comparison on the BindingDB and BioSNAP datasets with random split, unseen drug, and unseen target settings (**Best**, Second Best).

Setting	DeepConv-DTI[8]	GraphDTA[9]	MolTrans[1]	DrugBAN
		BindingDB		
Random Split	0.945±0.002	0.951±0.002	0.952±0.002	**0.960±0.001**
Unseen Drug	0.943±0.004	0.950±0.004	0.945±0.004	**0.959±0.002**
Unseen Target	0.627±0.070	0.670±0.023	0.661±0.037	**0.692±0.038**
		BioSNAP		
Random Split	0.886±0.006	0.887±0.008	0.895±0.004	**0.903±0.005**
Unseen Drug	0.856±0.005	0.858±0.007	0.856±0.008	**0.886±0.005**
Unseen Target	0.692±0.017	0.704±0.010	**0.714±0.014**	0.710±0.016

independent runs. Table 6 presents the AUROC results on the test sets, including the results on the usual random split for comparison. DrugBAN achieves the best performance in five of the six settings, while its performance in the unseen target setting of BioSNAP is also very competitive.

We need to point out that the model performance under the unseen drug setting only dropped slightly compared to that under the random split for all methods on BindingDB. This is because there are many highly similar molecules in the DTI datasets, and naive unseen drug setting does not distinguish them. A better strategy is the clustering-based split strategy in our previous study to alleviate this issue, leading to a more challenging cross-domain task.

6 Performance comparison with high fraction of missing data

Table 7. Performance comparison (average AUROC over five random runs) on the BindingDB and BioSNAP datasets with high fraction of missing data (**Best**, Second Best)

Missing (%)	DeepConv-DTI[8]	GraphDTA[9]	MolTrans[1]	DrugBAN
		BindingDB		
95	0.773±0.005	0.831±0.002	0.846±0.004	**0.856±0.003**
90	0.840±0.002	0.867±0.002	0.874±0.003	**0.887±0.004**
80	0.877±0.002	0.897±0.003	0.905±0.001	**0.920±0.003**
70	0.890±0.005	0.916±0.002	0.923±0.001	**0.934±0.001**
		BioSNAP		
95	0.710±0.005	0.768±0.005	0.767±0.006	**0.770±0.008**
90	0.781±0.003	0.798±0.003	0.800±0.004	**0.802±0.003**
80	0.816±0.003	0.829±0.003	0.835±0.001	**0.836±0.002**
70	0.839±0.002	0.851±0.002	0.853±0.002	**0.860±0.003**

We conduct experiments to clarify how the proposed model performs with high fraction of missing data on BindingDB and BioSNAP. Following the missing data setting in MolTrans[1], we train DrugBAN and deep learning baselines with only 5%, 10%, 20% and 30% of each dataset, and evaluate predictive performance on the rest of data (90% as test set and 10% as validation set for determining early stopping). Table 7 presents the obtained results, showing DrugBAN has the best performance in all settings. In particular, the improvement is larger on the bigger dataset (BindingDB).

7 Scalability

We study the scalability of DrugBAN from three different perspectives: model optimization time, data loading time and GPU memory usage. We use the default hyperparameter configuration in Table 4, and a single Nvidia V100 GPU to train the model in 100 epochs. Figure 3a illustrates the model optimization time and data loading time against the number of DTI pairs for 4,919 (10%) - 49,199 (100%) from the BindingDB dataset. We empirically observe that the optimization time (red line) of DrugBAN increases almost linearly with the number of DTI pairs. It takes about two hours for 49,199 DTI pairs to complete the optimization. The data loading process (blue line) takes more time than model optimization. Nevertheless, since the data loading can be done on CPU, we can accelerate the process with multiple loading workers (subprocesses) in parallel. Figure 3b shows the data loading time changes with respect to the number of workers, and it reduces significantly with only two additional

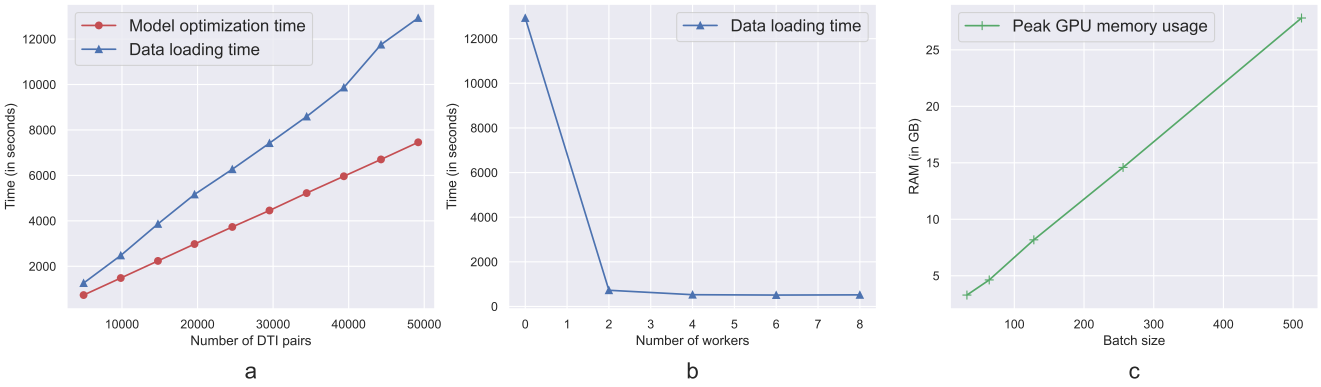

Figure 3. **Scalability of DrugBAN on the BindingDB dataset** (**a**) Model optimization and data loading time increase almost linearly with the number of DTI pairs. (**b**) Data loading time significantly reduces with the increasing number of workers. (**c**) Peak GPU memory usage increases linearly with the batch size.

workers added. Figure 3c shows the peak GPU memory usage against the batch size. We find that DrugBAN only takes up 4.63 GB RAM with the default batch size 64, which is highly efficient. Similar to the optimization time, the memory usage also increases linearly with the batch size. This study demonstrates the scalability of DrugBAN.

References

1. Huang, K., Xiao, C., Glass, L. & Sun, J. MolTrans: Molecular interaction transformer for drug–target interaction prediction. *Bioinformatics* **37**, 830 – 836 (2021).

2. Liu, H., Sun, J., Guan, J., Zheng, J. & Zhou, S. Improving compound–protein interaction prediction by building up highly credible negative samples. *Bioinformatics* **31**, i221 – i229 (2015).

3. Gilson, M. K. *et al.* BindingDB in 2015: a public database for medicinal chemistry, computational chemistry and systems pharmacology. *Nucleic acids research* **44**, D1045–D1053 (2016).

4. Bai, P. *et al.* Hierarchical clustering split for low-bias evaluation of drug-target interaction prediction. *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)* 641–644 (2021).

5. Gao, K. Y. *et al.* Interpretable drug target prediction using deep neural representation. In *IJCAI*, 3371–3377 (2018).

6. Wang, Z., Liang, L., Yin, Z. & Lin, J. Improving chemical similarity ensemble approach in target prediction. *Journal of cheminformatics* **8**, 1–10 (2016).

7. Yamanishi, Y., Araki, M., Gutteridge, A., Honda, W. & Kanehisa, M. Prediction of drug–target interaction networks from the integration of chemical and genomic spaces. *Bioinformatics* **24**, i232 – i240 (2008).

8. Lee, I., Keum, J. & Nam, H. DeepConv-DTI: Prediction of drug-target interactions via deep learning with convolution on protein sequences. *PLoS Computational Biology* **15** (2019).

9. Nguyen, T. *et al.* GraphDTA: Predicting drug-target binding affinity with graph neural networks. *Bioinformatics* **37**, 1140–1147 (2021).