nature machine intelligence

Article

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

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

Received: 1 August 2022

Accepted: 22 December 2022

Published online: 2 February 2023

Check for updates

Peizhen Bai[1], Filip Miljković[2], Bino John[3] & Haiping Lu[1]

Predicting drug–target interaction is key for drug discovery. Recent deep learning-based methods show promising performance, but two challenges remain: how to explicitly model and learn local interactions between drugs and targets for better prediction and interpretation and how to optimize generalization performance of predictions on novel drug–target pairs. Here, we present DrugBAN, a deep bilinear attention network (BAN) framework with domain adaptation to explicitly learn pairwise local interactions between drugs and targets, and adapt in response to out-of-distribution data. DrugBAN works on drug molecular graphs and target protein sequences to perform prediction, with conditional domain adversarial learning to align learned interaction representations across different distributions for better generalization on novel drug–target pairs. Experiments on three benchmark datasets under both in-domain and cross-domain settings show that DrugBAN achieves the best overall performance against five state-of-the-art baseline models. Moreover, visualizing the learned bilinear attention map provides interpretable insights from prediction results.

Drug–target interaction (DTI) prediction serves as an important step in the process of drug discovery[1–3]. Traditional biomedical measurement from in vitro experiments is reliable but has notably high cost and time-consuming development cycles, preventing its application to large-scale data[4]. By contrast, identifying high-confidence DTI pairs by in silico approaches can greatly narrow down the search scope of compound candidates, and provide insights into the causes of potential side effects in drug combinations. Therefore, in silico approaches have gained increasing attention and made much progress in the past few years[5,6].

For in silico approaches, traditional structure-based and ligand-based virtual screening methods have been studied widely for their relatively effective performance[7]. However, structure-based virtual screening requires molecular docking simulation, which is not applicable if the target protein's three-dimensional (3D) structure is unknown. Furthermore, ligand-based virtual screening predicts new active molecules based on the known actives of the same protein, but the performance is poor when the number of known actives is insufficient[8].

More recently, deep learning-based approaches have rapidly progressed for computational DTI prediction due to their successes in other areas, enabling large-scale validation in a relatively short time[9]. Many of them are constructed from a chemogenomics perspective[3,10], which integrates the chemical space, genomic space and interaction information into a unified end-to-end framework. As the number of biological targets that have available 3D structures is limited, many deep learning-based models take linear or two-dimensional (2D) structural information of drugs and proteins as inputs. They treat DTI prediction as a binary classification task, and make predictions by feeding the inputs into different deep encoding and decoding modules such

[1]Department of Computer Science, University of Sheffield, Sheffield, UK. [2]Imaging and Data Analytics, Clinical Pharmacology & Safety Sciences, R&D, AstraZeneca, Gothenburg, Sweden. [3]Imaging and Data Analytics, Clinical Pharmacology & Safety Sciences, R&D, AstraZeneca, Waltham, USA.
e-mail: h.lu@sheffield.ac.uk

as deep neural network (DNN)[11,12], graph neural network (GNN)[9,13–15] or transformer architectures[16,17]. With the advances of deep learning techniques, such models can automatically learn data-driven representations of drugs and proteins from large-scale DTI data instead of using only pre-defined descriptors.

Despite these promising developments, two challenges remain for existing deep learning-based methods. The first challenge is explicit learning of interactions between local structures of drug and protein. DTI is essentially decided by mutual effects between important molecular substructures in the drug compound and binding sites in the protein sequence[18]. However, many previous models learn global representations using their separate encoders, without explicitly learning local interactions[2,11,13,19,20]. Consequently, drug and protein representations are learned for the whole structures first, and mutual information is only implicitly learned in the black-box decoding module. Interactions between drug and target are particularly related to their crucial substructures; therefore, separate global representation learning tends to limit the modelling capacity and prediction performance. Moreover, without explicit learning of local interactions, the prediction result is hard to interpret, even if the prediction is accurate.

The second challenge is generalizing prediction performance across domains, beyond the learned distribution. Owing to the vast regions of chemical and genomic space, drug–target pairs that need to be predicted in real-world applications are often unseen and dissimilar to any pairs in the training data. They have different distributions and therefore need cross-domain modelling[21,22]. A robust model should be able to transfer learned knowledge to a new domain that only has unlabelled data. In this case, we need to align distributions and improve cross-domain generalization performance by learning transferable representations; for example, from 'source' to 'target'. To our knowledge, this is an underexplored direction in drug discovery[23].

To address these challenges, we propose an interpretable bilinear attention network-based model (DrugBAN) for DTI prediction, as shown in Fig. 1a. DrugBAN is a deep learning framework with explicit learning of local interactions between drug and target, and conditional domain adaptation for learning transferable representations across domains. Specifically, we first use graph convolutional networks[24] (GCNs) and convolutional neural networks (CNNs) to encode local structures as a 2D molecular graph and one-dimensional (1D) protein sequence. Then the encoded local representations are fed into a pairwise interaction module that consists of a bilinear attention network[25,26] to learn local interaction representations, as depicted in Fig. 1b. The local joint interaction representations are decoded by a fully connected layer to make a DTI prediction. In this way, we can utilize the pairwise bilinear attention map to visualize the contribution of each substructure to the final predictive result, improving the interpretability. For cross-domain prediction, we apply conditional domain adversarial network[27] (CDAN) to transfer learned knowledge from source domain to target domain to enhance cross-domain generalization, as illustrated in Fig. 1c. We conduct a comprehensive performance comparison against five state-of-the-art DTI prediction methods on both in-domain and cross-domain settings of drug discovery. The results show that our method achieves the best overall performance compared to state-of-the-art methods, while providing interpretable insights for the prediction results.

To summarize, DrugBAN differs from previous works in three main ways. First, it captures pairwise local interactions between drugs and targets with a bilinear attention mechanism. Second, it enhances cross-domain generalization with an adversarial domain adaptation approach. It gives an interpretable prediction with bilinear attention weights instead of black-box results.

Results

Problem formulation

In DTI prediction, the task is to determine whether a pair of a drug compound and a target protein will interact. For the target protein, we denote each protein sequence as $\mathcal{P} = (a_1, ..., a_n)$, where each token a_i represents one of the 23 amino acids. For the drug compound, most existing deep learning-based methods represent the input by the simplified molecular-input line-entry system (SMILES)[28], which is a 1D sequence describing chemical atom and bond token information in the drug molecule. The SMILES format enables encoding of drug information with many classic deep learning architectures. However, given that the 1D sequence is not a natural representation for molecules, some important structural information of drugs could be lost, degrading model prediction performance. Our model converts input SMILES into its corresponding 2D molecular graph. Specifically, a drug molecular graph is defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertices (atoms) and $\mathcal{E}$ is the set of edges (chemical bonds).

Given a protein sequence $\mathcal{P}$ and a drug molecular graph $\mathcal{G}$, DTI prediction aims to learn a model $\mathcal{M}$ to map the joint feature representation space $\mathcal{P} \times \mathcal{G}$ to an interaction probability score $p \in [0, 1]$. Supplementary Table 3 provides the commonly used notations in this paper.

DrugBAN framework

The proposed DrugBAN framework is shown in Figure 1a. Given an input drug–target pair, we first use separate GCN and 1D convolutional neural network (1D CNN) blocks to encode molecular graph and protein sequence information, respectively. Then we use a bilinear attention network module to learn local interactions between encoded drug and protein representations. The bilinear attention network consists of a bilinear attention step and a bilinear pooling step to generate a joint representation, as illustrated in Fig. 1b. Second, a fully connected classification layer learns a predictive score, indicating the probability of interaction. To improve model generalization performance on cross-domain drug–target pairs, we further embed CDAN into the framework to adapt representations for better aligning source and target distributions, as shown in Fig. 1c.

Evaluation strategies and metrics

We study classification performance on three public datasets separately: BindingDB[29], BioSNAP[30] and Human[16,31], with hold-out test sets ('unknown') kept back for evaluation. We use two different split strategies for in-domain and cross-domain settings. For in-domain evaluation, each experimental dataset is randomly divided into training, validation and test sets with a 7:1:2 ratio. For cross-domain evaluation, we propose a clustering-based pair split strategy to construct cross-domain scenarios. We conduct cross-domain evaluation on the large-scale BindingDB and BioSNAP datasets. For each dataset, we first use the single-linkage algorithm to cluster drugs and proteins by ECFP4 (extended connectivity fingerprint, up to four bonds)[32] fingerprint and pseudo-amino acid composition (PSC)[33], respectively. After that, we randomly select 60% drug clusters and 60% protein clusters from the clustering result, and consider all drug–target pairs between the selected drugs and proteins as source domain data. All the pairs between drugs and proteins in the remaining clusters are considered to be target domain data. The clustering implementation details are provided in the Supplementary Information, section 1. Under the clustering-based pair split strategy, the source and target domains are non-overlapping with different distributions. Following the general setting of domain adaptation, we use all labelled source domain data and 80% unlabelled target domain data as the training set, and the remaining 20% labelled target domain data as the test set. The cross-domain evaluation is more challenging than in-domain random split, but provides a better measure of model generalization ability in real-world drug discovery. For a more comprehensive study, we report additional experiments across different protein families, on unseen drugs or targets, and with those with a high fraction of missing data (Supplementary Information, sections 4–6, respectively).

The area under the receiver operating characteristic curve (AUROC) and the area under the precision-recall curve (AUPRC) are

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

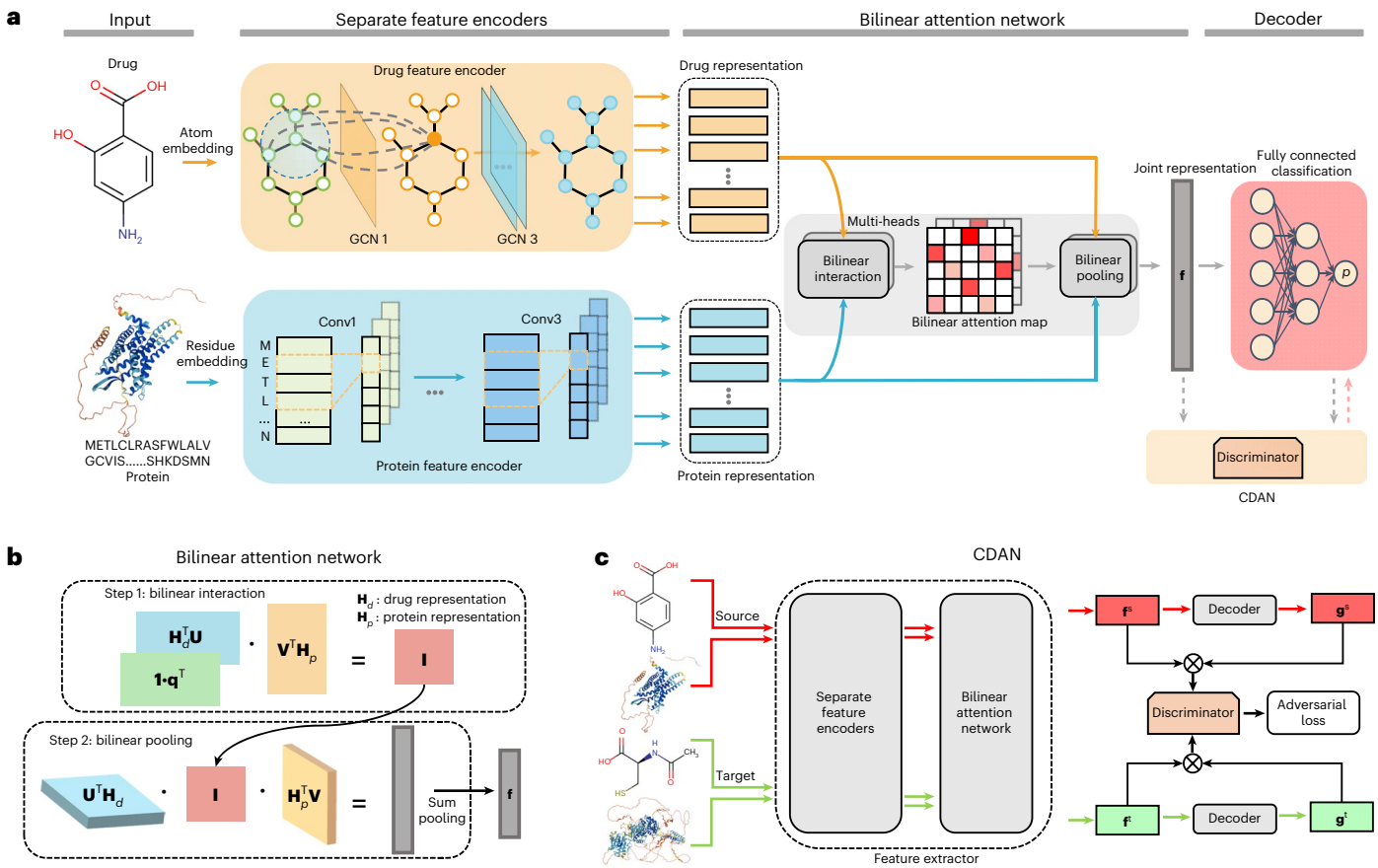

Fig. 1 | Overview of the DrugBAN framework. a, The input drug molecule and protein sequence are separately encoded by GCNs and 1D CNNs. Each row of the encoded drug representation is an aggregated representation of adjacent atoms in the drug molecule, and each row of the encoded protein representation is a subsequence representation in the protein sequence. The drug and protein representations are fed into a bilinear attention network to learn their pairwise local interactions. The joint representation **f** is decoded by a fully connected decoder module to predict the DTI probability *p*. If the prediction task is cross-domain, the CDAN[27] module is employed to align learned representations in the source and target domains. **b**, The bilinear attention network architecture. $\mathbf{H}_d$ and $\mathbf{H}_p$ are encoded drug and protein representations. In step 1, the bilinear attention map matrix **I** is obtained by a low-rank bilinear interaction modelling through transformation matrices **U** and **V** to measure the substructure-level interaction intensity[59]. Then **I** is utilized to produce the joint representation **f** in step 2 by bilinear pooling by the shared transformation matrices **U** and **V**. **c**, CDAN is a domain adaptation technique to reduce the domain shift between different distributions of data. We use CDAN to embed joint representation **f** and softmax logits **g** for source and target domains into a joint conditional representation by the discriminator, a two-layer fully connected network that minimizes the domain classification error to distinguish the target domain from the source domain.

used as the major metrics to evaluate model classification performance. In addition, we also report accuracy, sensitivity and specificity at the threshold of the best F1 score. We conduct five independent runs with different random seeds for each dataset split. The best performing model is the one with the best AUROC on the validation set. This model is then evaluated on the test set to report the performance metrics.

In-domain performance comparison

Here, we compare DrugBAN with five baselines under the random split setting: support vector machine (SVM)[34], random forest (RF)[35], DeepConv-DTI[11], GraphDTA[13] and MolTrans[17]. This is the in-domain scenario, so we use vanilla DrugBAN without embedding the CDAN module. Table 1 shows the comparison on the BindingDB and BioSNAP datasets. DrugBAN has consistently outperformed baselines in terms of AUROC, AUPRC and accuracy, while its performance in sensitivity and specificity is also competitive. The results indicate that data-driven representation learning can capture more important information than pre-defined descriptor features in in-domain DTI prediction. Moreover, DrugBAN can capture interaction patterns through its pairwise interaction module, further improving prediction performance.

The in-domain results on the Human dataset are shown in Figure 2. Under the random split, the deep learning-based models all achieve similar and promising performance (AUROC > 0.98). However, as pointed out in ref. [16], the Human dataset had some hidden ligand bias, resulting in the correct predictions being made only based on the drug features rather than interaction patterns. The high accuracy could be due to bias and overfitting, not a model's real-world performance of prospective prediction. Therefore, we further use a cold pair split strategy to evaluate models to mitigate the overoptimism of performance estimation under random split due to the data bias. This cold pair split strategy guarantees that all test drugs and proteins are not observed during training so that prediction on test data cannot rely only on the features of known drugs or proteins. We randomly assign 5% and 10% DTI pairs into the validation and test sets, respectively, and remove all of their associated drugs and proteins from the training set. Figure 2 indicates that all models have a significant performance drop from random split to cold pair split, especially for SVM and RF. However, we can see that DrugBAN still achieves the best performance against other state-of-the-art deep learning baselines.

Cross-domain performance comparison

In-domain classification under random split is an easier task and of less practical importance. Therefore, next, we study more realistic and challenging cross-domain DTI prediction, in which training data

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

Table 1 | In-domain performance comparison on the BindingDB and BioSNAP datasets with random split (statistics over five random runs)

Dataset	Method	AUROC	AUPRC	Accuracy	Sensitivity	Specificity
BindingDB	SVM[34]	0.939±0.001	0.928±0.002	0.825±0.004	0.781±0.014	0.886±0.012
	RF[35]	0.942±0.011	0.921±0.016	0.880±0.012	0.875±0.023	0.892±0.020
	DeepConv-DTI[11]	0.945±0.002	0.925±0.005	0.882±0.007	0.873±0.018	0.894±0.009
	GraphDTA[13]	0.951±0.002	0.934±0.002	0.888±0.005	0.882±0.012	0.897±0.008
	MolTrans[17]	0.952±0.002	0.936±0.001	0.887±0.006	0.877±0.016	0.902±0.009
	DrugBAN	**0.960±0.001**	**0.948±0.002**	**0.904±0.004**	**0.900±0.008**	**0.908±0.004**
BioSNAP	SVM[34]	0.862±0.007	0.864±0.004	0.777±0.011	0.711±0.042	0.841±0.028
	RF[35]	0.860±0.005	0.886±0.005	0.804±0.005	**0.823±0.032**	0.786±0.025
	DeepConv-DTI[11]	0.886±0.006	0.890±0.006	0.805±0.009	0.760±0.029	0.851±0.013
	GraphDTA[13]	0.887±0.008	0.890±0.007	0.800±0.007	0.745±0.032	**0.854±0.025**
	MolTrans[17]	0.895±0.004	0.897±0.005	0.825±0.010	0.818±0.031	0.831±0.013
	DrugBAN	**0.903±0.005**	**0.902±0.004**	**0.834±0.008**	0.820±0.021	0.847±0.010

The results are presented as mean±standard deviation. The best result for each dataset and metric is marked in bold and the second-best result is underlined.

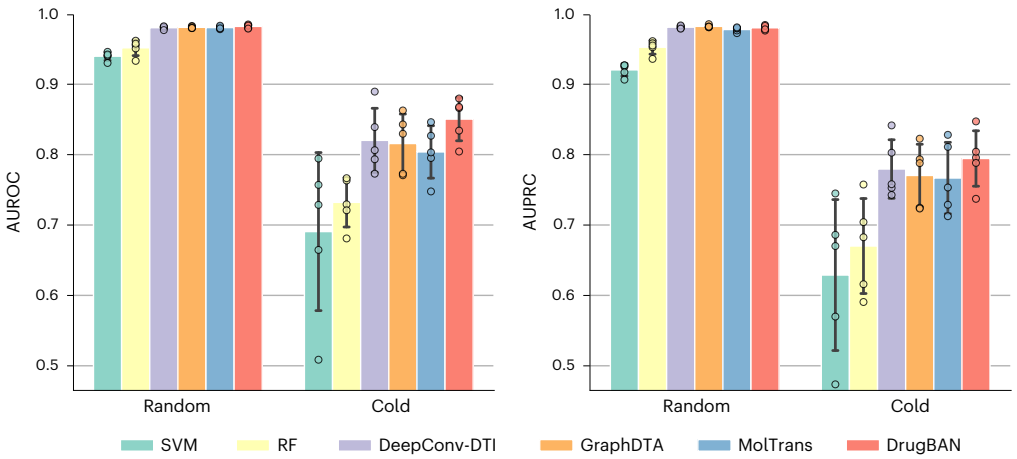

Fig. 2 | In-domain performance comparison on the Human dataset with random split and cold pair split (statistics over five random runs). The vertical bars represent the mean, and the black lines are error bars indicating the standard deviation. The dots indicates performance scores in each random run of models. Supplementary Table 2 provides the data statistics of the Human dataset.

and test data have different distributions. To imitate this scenario, the original data are divided into source and target domains by the clustering-based pair split. We turn on the CDAN module of DrugBAN (that is, we use DrugBAN$_{CDAN}$) to study knowledge transferability in cross-domain prediction.

The performance evaluation on the BindingDB and BioSNAP datasets with clustering-based pair split is presented in Figure 3. Compared to the previous in-domain prediction results, the performance of all DTI models drops significantly due to less information overlap between training and test data. In this scenario, vanilla DrugBAN still outperforms other state-of-the-art models on the whole. Specifically, it outperforms MolTrans by 2.9% and 7.4% in AUROC on the BioSNAP and BindingDB datasets, respectively. The results show that DrugBAN is a robust method under both in-domain and cross-domain settings. Interestingly, RF achieves good performance and even consistently outperforms other deep learning baselines (DeepConv, GraphDTA and MolTrans) on the BindingDB dataset. The results indicate that deep learning methods are not always superior to shallow machine learning methods under the cross-domain setting.

Recently, domain adaptation techniques have received increasing attention due to their ability to transfer knowledge across domains, but they are mainly applied to computer vision and natural language processing problems. We combine vanilla DrugBAN with CDAN to tackle cross-domain DTI prediction. As shown in Fig. 3, DrugBAN$_{CDAN}$ has significant performance improvements with the introduction of a domain adaptation module. On the BioSNAP dataset, it outperforms vanilla DrugBAN by 4.6% and 16.9% in AUROC and AUPRC, respectively. By minimizing the distribution discrepancy across domains, CDAN can effectively enhance DrugBAN generalization ability and provide more reliable results.

These results demonstrate the strength of DrugBAN in generalizing prediction performance across domains.

Ablation study

Here, we conduct an ablation study to investigate the influences of bilinear attention and domain adaptation modules on DrugBAN. The results are shown in Table 2. To validate the effectiveness of bilinear attention, we study three variants of DrugBAN that differ in the joint representation computation between drug and protein: one-side drug

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

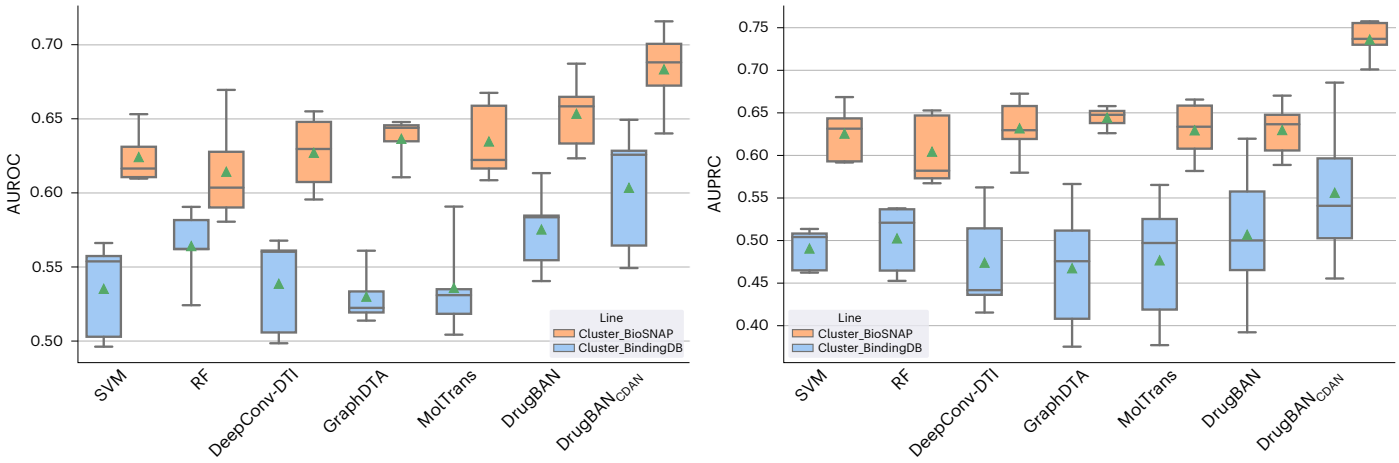

Fig. 3 | Cross-domain performance comparison on the BindingDB and BioSNAP datasets with clustering-based pair split (statistics over five random runs). The box plots show the median as the centre lines and the mean as green triangles. The minima and lower percentile represent the worst and second-worst scores. The maxima and upper percentile indicate the best and second-best scores. Supplementary Table 2 provides the data statistics of the BindingDB and BioSNAP datasets.

Table 2 | Ablation study in AUROC on the BindingDB and BioSNAP datasets with random split and clustering-based split strategies (statistics over five random runs)

Ablation tests	BindingDB$_{random}$	BioSNAP$_{random}$	BindingDB$_{cluster}$	BioSNAP$_{cluster}$
Linear concatenation[2,11,13]	0.949±0.002	0.887±0.007	NA	NA
One-side target attention[14]	0.950±0.002	0.890±0.005	NA	NA
One-side drug attention[14]	0.953±0.002	0.892±0.004	NA	NA
DrugBAN	**0.960±0.001**	**0.903±0.005**	0.575±0.025	0.654±0.023
MolTrans$_{CDAN}$	NA	NA	0.575±0.038	0.656±0.028
DrugBAN$_{DANN}$	NA	NA	0.592±0.042	0.667±0.030
DrugBAN$_{CDAN}$	NA	NA	**0.604±0.039**	**0.684±0.026**

The results are presented as mean±standard deviation. The first four models show the effectiveness of our bilinear attention module, and the last three models show the strength of DrugBAN$_{CDAN}$ on cross-domain prediction. The best AUROC result for each dataset is marked in bold and the second-best result is underlined. NA, not applicable to this study.

attention, one-side protein attention and linear concatenation. The one-side attention is equivalent to the neural attention mechanism introduced in ref. [14], which is used to capture the joint representation between a drug vector representation and a protein subsequence matrix representation. We replace the bilinear attention in DrugBAN with one-side attention to generate the two variants. Linear concatenation is a simple vector concatenation of drug and protein vector representations after a max-pooling layer. As shown in the first four rows of Table 2, the results demonstrate that bilinear attention is the most effective method to capture interaction information for DTI prediction. To examine the effect of CDAN, we study two variants: DrugBAN with domain adversarial neural network (DANN)[36] (that is, DrugBAN$_{DANN}$) and MolTrans with CDAN (that is, MolTrans$_{CDAN}$). DANN is another adversarial domain adaptation technique that does not take into consideration classification distribution. The last four rows of Table 2 indicate that DrugBAN$_{CDAN}$ still achieves the best performance improvement in cross-domain prediction.

Interpretability with bilinear attention visualization

A further strength of DrugBAN is to enable molecular level insights and interpretation critical for drug design efforts, utilizing the components of the bilinear attention map to visualize the contribution of each substructure to the final predictive result. Here, we examine the top three predictions (PDB IDs: 6QL2 (ref. [37]), 5W8L (ref. [38]) and 4N6H (ref. [39])) of co-crystalized ligands from the Protein Data Bank (PDB)[40]. Only X-ray structures with resolution greater than 2.5 Å that corresponded to human protein targets proceeded to selection. In addition, co-crystalized ligands were required to have $pIC_{50} \leq 100$ nM and not to be part of the training set. The visualization results are shown in Fig. 4a alongside the ligand–protein interaction maps originating from the corresponding X-ray structures. For each molecule, we coloured its top 20% weighted atoms in the bilinear attention map in orange.

For PDB structure 6QL2 (ethoxzolamide complexed with human carbonic anhydrase 2), our model correctly interpreted the sulfonamide region as essential for ligand–protein binding (with sulfonamide oxygen as a hydrogen bond acceptor to the backbone of Leu198 and Thr199, and the amino group as a hydrogen bond donor to the side chains of His94 and Thr199). Conversely, the ethoxy group of ethoxzolamide was incorrectly predicted to form specific interactions with the protein, although its exposure to the solvent may promote further binding (blue highlight). In addition, benzothiazole scaffold, which forms an arene–H interaction with Leu198, is only partly highlighted by our interpretability model. It is worth mentioning that although the top 20% of interacting atoms of ethoxzolamide corresponded to only three highlighted atoms, all of them indicated different ligand–protein interaction sites corroborated by the X-ray structure.

In structure 5W8L (9YA ligand bound to human l-lactate dehydrogenase A), the interpretability feature once more highlighted important interaction patterns for ligand–protein binding. For example, the sulfonamide group was once more indicated to form specific

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

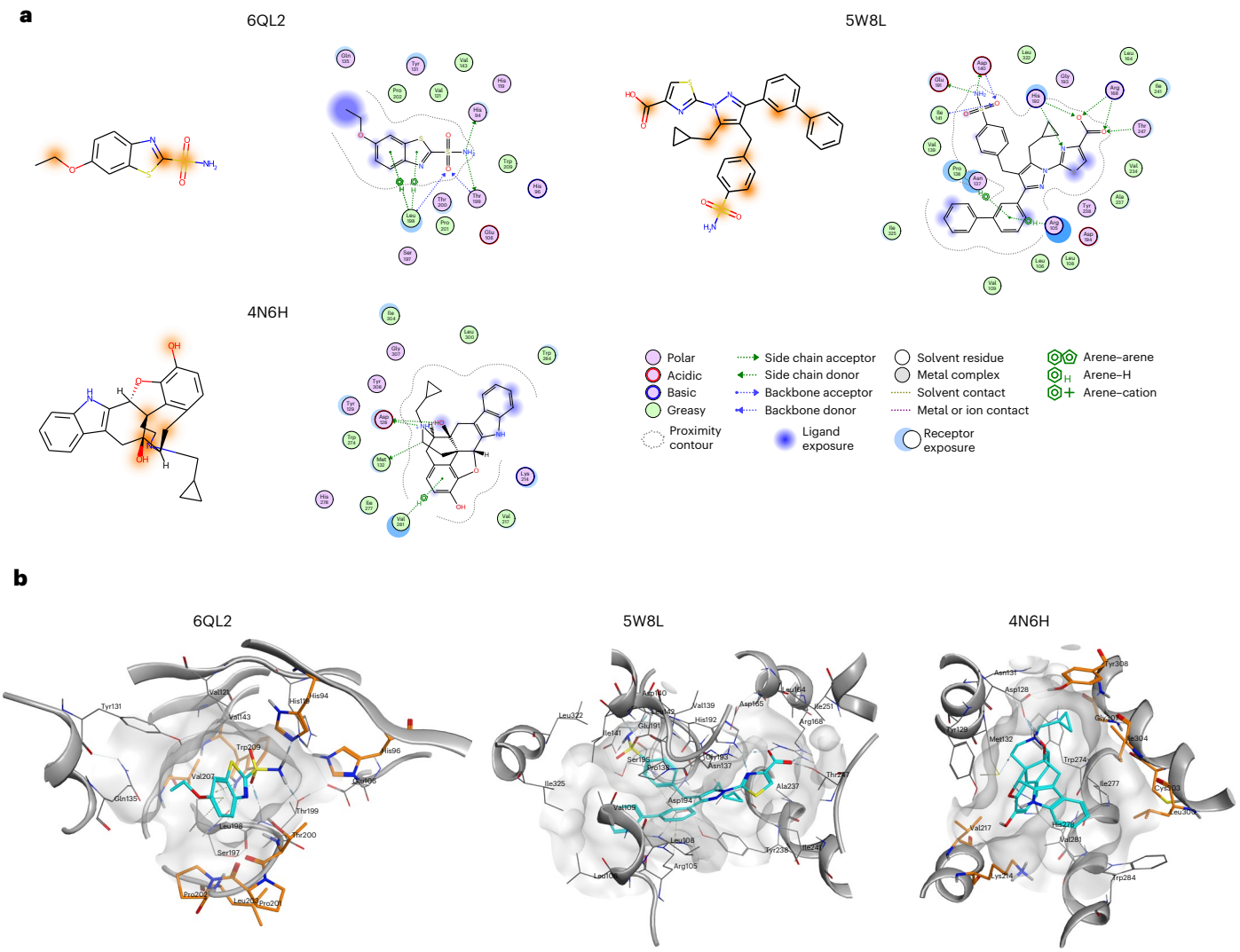

Fig. 4 | Visualization of ligands and binding pockets for interpretability study. a, Interpretability of co-crystalized ligands. The left side of each panel shows the 2D structures of ligands with highlighted atoms (orange) that were predicted to contribute to protein binding. All structures were visualized using RDKit[56]. In addition, ligand–protein interaction maps (right side of each panel) from the corresponding crystal structures of these ligands are provided. **b**, Interpretability of binding pocket structures. The 3D representations of ligand–protein binding pockets are provided, highlighting the correctly predicted amino acid residues (orange) that surround the corresponding ligands (cyan). The remaining amino acid residues, secondary structure elements and surface maps are coloured in grey. All ligand–protein interaction maps and 3D representations of X-ray structures were visualized using the Molecular Operating Environment (MOE) software.

interactions with the protein (with the amino group as a hydrogen bond donor to the side chains of Asp140 and Glu191, and sulfonamide oxygen as a hydrogen bond acceptor to the backbone of Asp140 and Ile141). Similarly, we noted that the carboxylic acid group was also partly highlighted (in 5W8L, carboxylic acid oxygens act as hydrogen bond acceptors to the side chains of Arg168, His192 and Thr247). Moreover, biphenyl rings were correctly predicted to participate in ligand–protein binding (in 5W8L, arene–H interaction with Arg105 and Asn137). Although 9YA (bound to 5W8L) was much larger and complex than ethoxzolamide (bound to 6QL2), the model showed good interpretability potential for the majority of the experimentally confirmed interactions.

In the third example, 4N6H X-ray complex of human delta-type opioid receptor with EJ4 ligand, the main interacting functional groups of EJ4 were once more highlighted correctly. Here, a hydroxyl group of the aliphatic ring complex and a neighbouring tertiary amine (both as hydrogen bond donors to the side chain of Asp128) were correctly interpreted to form specific interactions. However, the phenol group was wrongly predicted to participate in protein binding.

As for the more challenging protein sequence interpretability, the results were weaker overall than those for ligand interpretability. Although many amino acid residues that were predicted to potentially participate in ligand binding were in fact distantly located to the respective compounds, a number of amino acid residues forming the binding sites were correctly predicted (Fig. 4b). For example, in 6QL2 complex, the following residues were highlighted: His94, His96, Thr200, Pro201, Pro202, Leu203, Val207 and Trp209. Among these, only His94 forms a specific interaction with ethoxzolamide. In 5W8L, none of the residues that constitute the ligand–protein binding site were highlighted. However, in 4N6H, there were several correctly predicted residues within the binding site: Lys214, Val217, Leu300, Cys303, Ile304, Gly307 and Tyr308. Unfortunately, none of the residues participated in the specific interactions with the ligand. Given these results, it is expected that protein sequence interpretability would be less confident because

the 1D protein sequence (used as protein information input in our model) does not necessarily indicate the 3D configuration and locality of the binding pocket. However, the results from the primary protein sequence are encouraging enough to safely assume that the further incorporation of 3D protein information into the modelling framework would eventually improve the model interpretability of drug–target interaction networks.

In addition, as the interpretability provided by DrugBAN is adaptively learned from DTI data itself, such interpretation has potential to find some hidden knowledge of local interactions that has not been explored, and could help drug hunters to improve binding properties of a given scaffold, or to reduce the off-target liabilities of a compound.

Conclusion

In this work, we present DrugBAN, an end-to-end bilinear attention deep learning framework for DTI prediction. We have integrated CDAN, an adversarial domain adaptation network, into the modelling process to enhance cross-domain generalization ability. Compared with other state-of-the-art DTI models and conventional machine learning models, the experimental results show that DrugBAN consistently achieves improved DTI prediction performance in both in-domain and cross-domain settings. Furthermore, by mapping attention weights to protein subsequences and drug compound atoms, our model can provide biological insights for interpreting the nature of interactions. The proposed ideas are general in nature and can be extended to other interaction prediction problems, such as the prediction of drug–drug interaction and protein–protein interaction.

This work focuses on chemogenomics-based DTI using a 1D protein sequence and 2D molecular graph as input. Given that the number of highly accurate 3D structured proteins accounts for only a small fraction of the known protein sequences, this work did not consider modelling with such structural information. Nevertheless, DeepMind's AlphaFold[41] is making great progress in protein 3D structure prediction, recently generating 2 billion protein 3D structure predictions from 1 million species. Such progress opens doors for utilizing 3D structural information in chemogenomics-based DTI prediction. Following the idea of pairwise local interaction learning and domain adaptation, we believe that extending our ideas further on complex 3D structures can lead to even better performance and interpretability in future work. Finally, this work studies different datasets separately; combining dataset integration with DrugBAN will be another interesting future direction to explore.

Methods

Bilinear attention network

This is an attention-based model and was first proposed to solve the problem of visual question answering (VQA)[26]. Given an image and relevant natural language question, VQA systems aim to provide a text–image matching answer. Therefore, VQA can be viewed as a multimodal learning task, similar to DTI prediction. Bilinear attention networks use a bilinear attention map to gracefully extend unitary attention networks to adapt to multimodal learning, by considering every pair of multimodal input channels (that is, the pairs of image regions and question words) to learn an interaction representation. Compared to using a unitary attention mechanism directly on multimodal data, bilinear attention networks can provide richer joint information but keep the computational cost at the same scale. Considering the similarity between VQA and DTI problems, we designed a bilinear attention network-inspired pairwise interaction module for DTI prediction.

Domain adaptation

These approaches train a model that reduces domain distribution shift between the source domain and target domain, which is mainly developed and studied in computer vision[42]. Early domain adaptation methods tended to reweight sample importance or learn invariant feature representations in shallow feature space, using labelled data in the source domain and unlabelled data in the target domain. More recently, deep domain adaptation methods embed the adaptation module in various deep architectures to learn transferable representations[43,44]. In particular, ref. [27] proposed a novel deep domain adaptation method, CDAN, that combines adversarial networks with multilinear conditioning for transferable representation learning. By introducing classifier prediction information into adversarial learning, CDAN can effectively align data distributions in different domains. We embed CDAN as an adaptation module in DrugBAN to enhance model performance for cross-domain DTI prediction.

DrugBAN architecture

CNN for protein sequence. The protein feature encoder consists of three consecutive 1D convolutional layers, which transform an input protein sequence to a matrix representation in the latent feature space. Each row of the matrix denotes a subsequence representation in the protein. Drawing on the concept of word embedding, we first initialize all amino acids into a learnable embedding matrix $E_p \in \mathbb{R}^{23\times D_p}$, where 23 is the number of amino acid types and D_p is the latent space dimensionality. By looking up E_p, each protein sequence $\mathcal{P}$ can be initialized to corresponding feature matrix $X_p \in \mathbb{R}^{\Theta_p\times D_p}$. Here, Θ_p is the maximum allowed length of a protein sequence, which is set to align different protein lengths and make batch training. Following previous studies[2,14,17], protein sequences with maximum allowed length are cut, and those with smaller length are padded with zeros.

The CNN-block protein encoder extracts local residue patterns from the protein feature matrix X_p. Here, a protein sequence is considered as an overlapping 3-mer amino acids such as METLCL... DSMN → MET, ETL, TLC,..., DSM, DLK. The first convolutional layer is utilized to capture the 3-mer residue-level features with kernel size = 3. Then the next two layers continue to enlarge the receptive field and learn more abstract features of local protein fragments. The protein encoder is described as follows:

$$\mathbf{H}_p^{(l+1)} = \sigma(\mathrm{CNN}(\mathbf{W}_c^{(l)}, \mathbf{b}_c^{(l)}, \mathbf{H}_p^{(l)})), \quad (1)$$

where $\mathbf{W}_c^{(l)}$ and $\mathbf{b}_c^{(l)}$ are the learnable weight matrices (filters) and bias vector in the lth CNN layer. $\mathbf{H}_p^{(l)}$ is the lth hidden protein representation and $\mathbf{H}_p^{(0)} = \mathbf{X}_p$, $\sigma(\cdot)$ denotes a non-linear activation function, with ReLU$(\cdot)$ used in our experiments.

GCN for molecular graph. For the drug compound, we convert each SMILES string to its 2D molecular graph $\mathcal{G}$. To represent node information in $\mathcal{G}$, we first initialize each atom node by its chemical properties, as implemented in the DGL-LifeSci[45] package. Each atom is represented as a 74-dimensional integer vector describing eight pieces of information: the atom type, the atom degree, the number of implicit Hs, the formal charge, the number of radical electrons, the atom hybridization, the number of total Hs and whether the atom is aromatic. Similar to the maximum allowed length setting in a protein sequence above, we set a maximum allowed number of nodes Θ_d. Molecules with less nodes will contain virtual nodes with zero padded. As a result, each graph's node feature matrix is denoted as $\mathbf{M}_d \in \mathbb{R}^{\Theta_d\times 74}$. Moreover, we use a simple linear transformation to define $\mathbf{X}_d = \mathbf{W}_0\mathbf{M}_d^\top$, leading to a real-valued dense matrix $\mathbf{X}_d \in \mathbb{R}^{\Theta_d\times D_d}$ as the input feature.

We use a three-layer GCN block to effectively learn the graph representation on drug compounds. GCN generalizes the convolutional operator to an irregular domain. Specifically, we update the atom feature vectors by aggregating their corresponding sets of neighbourhood atoms, connected by chemical bonds. This propagation mechanism automatically captures substructure information of a molecule. We keep the node-level drug representation for subsequent explicit learning of local interactions with protein fragments. The drug encoder is written as

$$\mathbf{H}_d^{(l+1)} = \sigma(\mathrm{GCN}(\tilde{\mathbf{A}}, \mathbf{W}_g^{(l)}, \mathbf{b}_g^{(l)}, \mathbf{H}_p^{(l)})), \quad (2)$$

where $\mathbf{W}_g^{(l)}$ and $\mathbf{b}_g^{(l)}$ are the layer-specific learnable weight matrix and bias vector of GCN, $\tilde{\mathbf{A}}$ is the adjacency matrix with added self-connections in molecular graph $\mathcal{G}$, and $\mathbf{H}_d^{(l)}$ is the lth hidden node representation with $\mathbf{H}_d^{(0)} = \mathbf{X}_d$.

Pairwise interaction learning. We apply a bilinear attention network module to capture pairwise local interactions between drug and protein. It consists of two layers: a bilinear interaction map to capture pairwise attention weights and a bilinear pooling layer over the interaction map to extract joint drug–target representation.

Separate CNN and GCN encoders in the third layer generate the hidden protein and drug representations $\mathbf{H}_p^{(3)} = \{\mathbf{h}_p^1, \mathbf{h}_p^2, ..., \mathbf{h}_p^M\}$ and $\mathbf{H}_d^{(3)} = \{\mathbf{h}_d^1, \mathbf{h}_d^2, ..., \mathbf{h}_d^N\}$, where M and N denote the number of encoded substructures in a protein and atoms in a drug. We construct the bilinear interaction map using these hidden representations to obtain a single head pairwise interaction matrix $\mathbf{I} \in \mathbb{R}^{N \times M}$:

$$\mathbf{I} = ((\mathbf{1} \cdot \mathbf{q}^\top) \circ \sigma((\mathbf{H}_d^{(3)})^\top \mathbf{U})) \cdot \sigma(\mathbf{V}^\top \mathbf{H}_p^{(3)}), \quad (3)$$

where $\mathbf{U} \in \mathbb{R}^{D_d \times K}$ and $\mathbf{V} \in \mathbb{R}^{D_p \times K}$ are learnable weight matrices for drug and protein representations, $\mathbf{q} \in \mathbb{R}^K$ is a learnable weight vector, $\mathbf{1} \in \mathbb{R}^N$ is a fixed all-ones vector and $\circ$ denotes Hadamard (element-wise) product. The elements in $\mathbf{I}$ indicate the interaction intensity of respective drug–target sub-structural pairs, with mapping to potential binding sites and molecular substructures. To intuitively understand bilinear interaction, an element $\mathbf{I}_{i,j}$ in equation (3) can also be written as

$$\mathbf{I}_{i,j} = \mathbf{q}^\top (\sigma(\mathbf{U}^\top \mathbf{h}_d^i) \circ \sigma(\mathbf{V}^\top \mathbf{h}_p^j)), \quad (4)$$

where $\mathbf{h}_d^i$ is the ith column of $\mathbf{H}_d^{(3)}$ and $\mathbf{h}_p^j$ is the jth column of $\mathbf{H}_p^{(3)}$, respectively, denoting the ith and jth sub-structural representations of drug and protein. Therefore, we can see a bilinear interaction as first mapping representations $\mathbf{h}_d^i$ and $\mathbf{h}_p^j$ to a common feature space with weight matrices $\mathbf{U}$ and $\mathbf{V}$, and then learn an interaction on Hadamard product and the weight of vector $\mathbf{q}$. In this way, pairwise interactions provide interpretability on the contribution of sub-structural pairs to the predicted result.

To obtain the joint representation $\mathbf{f}' \in \mathbb{R}^K$, we introduce a bilinear pooling layer over the interaction map $\mathbf{I}$. Specifically, the kth element of $\mathbf{f}'$ is computed as

$$\begin{aligned} \mathbf{f}'_k &= \sigma((\mathbf{H}_d^{(3)})^\top \mathbf{U})_k^\top \cdot \mathbf{I} \cdot \sigma((\mathbf{H}_p^{(3)})^\top \mathbf{V})_k \\ &= \sum_{i=1}^{N} \sum_{j=1}^{M} \mathbf{I}_{i,j} (\mathbf{h}_d^i)^\top (\mathbf{U}_k \mathbf{V}_k^\top) \mathbf{h}_p^j, \end{aligned} \quad (5)$$

where $\mathbf{U}_k$ and $\mathbf{V}_k$ denote the kth column of weight matrices $\mathbf{U}$ and $\mathbf{V}$. Notably, there are no new learnable parameters at this layer. The weight matrices $\mathbf{U}$ and $\mathbf{V}$ are shared with the previous interaction map layer to decrease the number of parameters and alleviate overfitting. Moreover, we add a sum pooling on the joint representation vector to obtain a compact feature map:

$$\mathbf{f} = \mathrm{SumPool}(\mathbf{f}', s), \quad (6)$$

where the $SumPool(\cdot)$ function is a 1D and non-overlapped sum pooling operation with stride s. It reduces the dimensionality of $\mathbf{f}' \in \mathbb{R}^K$ to $\mathbf{f} \in \mathbb{R}^{K/s}$. Furthermore, we can extend the single pairwise interaction to a multi-head form by calculating multiple bilinear interaction maps. The final joint representation vector is a sum of individual heads. As the weight matrices $\mathbf{U}$ and $\mathbf{V}$ are shared, each additional head only adds one new weight vector $\mathbf{q}$, which is parameter-efficient. In our experiments, the multi-head interaction has better performance than a single one.

Thus, using the novel bilinear attention mechanism, the model can explicitly learn pairwise local interactions between drug and protein. This interaction module is inspired by and adapted from refs. [26,25], in which two bilinear models are designed for the VQA problem. To compute the interaction probability, we feed the joint representation $\mathbf{f}$ into the decoder, which is one fully connected classification layer followed by a sigmoid function:

$$p = \mathrm{Sigmoid}(\mathbf{W}_o \mathbf{f} + \mathbf{b}_o), \quad (7)$$

where $\mathbf{W}_o$ and $\mathbf{b}_o$ are learnable weight matrix and bias vector.

Finally, we jointly optimize all learnable parameters by backpropagation. The training objective is to minimize the cross-entropy loss as follows:

$$\mathcal{L} = -\sum_i (y_i \log(p_i) + (1 - y_i) \log(1 - p_i)) + \frac{\lambda}{2} \|\mathbf{\Theta}\|_2^2, \quad (8)$$

where $\mathbf{\Theta}$ is the set of all learnable weight matrices and bias vectors above, y_i is the ground-truth label of the ith drug–target pair, p_i is its output probability by the model and λ is a hyperparameter for L2 regularization.

Cross-domain adaptation for better generalization. Machine learning models tend to perform well on similar data from the same distribution (that is, in-domain), but poorer on dissimilar data with different distribution (that is, cross-domain). It is a key challenge to improve model performance on cross-domain DTI prediction. In our framework, we embed CDAN to enhance generalization from a source domain with sufficient labelled data to a target domain for which only unlabelled data are available.

Given a source domain $\mathcal{S}_s = \{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ of N_s labelled drug–target pairs and a target domain $\mathcal{S}_t = \{x_i^t\}_{i=1}^{N_t}$ of N_t unlabelled drug–target pairs, we leverage CDAN to align their distributions and improve prediction performance across domains. Figure 1c shows the CDAN workflow in our framework, including three key components: the feature extractor $F(\cdot)$, the decoder $G(\cdot)$ and the domain discriminator $D(\cdot)$. We use $F(\cdot)$ to denote the separate feature encoders and bilinear attention network together to generate joint representations of input domain data; that is, $\mathbf{f}_i^s = F(x_i^s)$ and $\mathbf{f}_j^t = F(x_j^t)$. Next, we use the fully connected classification layer mentioned above followed by a softmax function as $G(\cdot)$ to obtain a classifier prediction $\mathbf{g}_i^s = G(\mathbf{f}_i^s) \in \mathbb{R}^2$ and $\mathbf{g}_j^t = G(\mathbf{f}_j^t) \in \mathbb{R}^2$. Furthermore, we apply a multilinear map to embed joint representation $\mathbf{f}$ and classifier prediction $\mathbf{g}$ into a joint conditional representation $\mathbf{h} \in \mathbb{R}^{2K/s}$, which is defined as the flattening of the outer product of the two vectors:

$$\mathbf{h} = \mathrm{FLATTEN}(\mathbf{f} \otimes \mathbf{g}), \quad (9)$$

where $\otimes$ is the outer product.

The multilinear map captures multiplicative interactions between two independent distributions[46,47]. Following the CDAN mechanism, we simultaneously align the joint representation and predicted classification distributions of source and target domains by conditioning the domain discriminator $D(\cdot)$ on the $\mathbf{h}$. The domain discriminator $D(\cdot)$, consisting of a three-layer fully connected networks, learns to distinguish whether a joint conditional representation $\mathbf{h}$ is derived from the source domain or the target domain. Conversely, the feature extractor $F(\cdot)$ and decoder $G(\cdot)$ are trained to minimize the source domain cross-entropy loss $\mathcal{L}$ with source label information, and simultaneously generate indistinguishable representation $\mathbf{h}$ to confuse the discriminator $D(\cdot)$. As a result, we can formulate the two losses in the cross-domain modelling:

$$\mathcal{L}_s(F, G) = \mathbb{E}_{(x_i^s, y_i^s) \sim \mathcal{S}_s} \mathcal{L}(G(F(x_i^s)), y_i^s), \quad (10)$$

$$\mathcal{L}_{adv}(F, G, D) = \mathbb{E}_{x_i^t \sim \mathcal{S}_t} \log(1 - D(\mathbf{f}_i^t, \mathbf{g}_i^t)) + \mathbb{E}_{x_j^s \sim \mathcal{S}_s} log(D(\mathbf{f}_j^s, \mathbf{g}_j^s)), \quad (11)$$

where $\mathcal{L}_s$ is the cross-entropy loss on the labelled source domain and $\mathcal{L}_{adv}$ is the adversarial loss for domain discrimination. The optimization problem is written as a minimax paradigm:

$$\max_{D} \min_{F,G} \mathcal{L}_s(F, G) - \omega \mathcal{L}_{adv}(F, G, D), \quad (12)$$

where $\omega > 0$ is a hyperparameter to weight $\mathcal{L}_{adv}$. By introducing the adversarial training on $\mathcal{L}_{adv}$, our framework can reduce the data distribution shift between source and target domains, leading to the improved generalization on cross-domain prediction.

Experimental setting

Datasets. We evaluate DrugBAN and five state-of-the-art baselines on three public DTI datasets: BindingDB, BioSNAP and Human. The BindingDB dataset is a web-accessible database[48] of experimentally validated binding affinities, focusing primarily on the interactions of small drug-like molecules and proteins. We use a low-bias version of the BindingDB dataset constructed in our earlier work (ref. [49]), with the bias-reducing preprocessing steps described in the Supplementary Information, section 2. The BioSNAP dataset is created from the DrugBank database[50] by ref. [17] and ref. [30], consisting of 4,510 drugs and 2,181 proteins. It is a balanced dataset with validated positive interactions and an equal number of negative samples randomly obtained from unseen pairs. The Human dataset is constructed by ref. [31], including highly credible negative samples by an in silico screening method. Following previous studies[14,16,20], we also use the balanced version of Human dataset containing the same number of positive and negative samples. To mitigate the influence of the hidden data bias[16], we use additional cold pair split for performance evaluation on the Human dataset. Supplementary Table 2 shows statistics of the three datasets.

Implementation. DrugBAN is implemented in Python 3.8 and PyTorch 1.7.1 (ref. [51]), along with functions from DGL 0.7.1 (ref. [52]), DGL-lifeSci 0.2.8 (ref. [45]), Scikit-learn 1.0.2 (ref. [53]), Numpy 1.20.2 (ref. [54]), Pandas 1.2.4 (ref. [55]) and RDKit 2021.03.2 (ref. [56]). The batch size is set to be 64 and the Adam optimizer is used with a learning rate of 5e-5. We allow the model to run for at most 100 epochs for all datasets. The best performing model is selected at the epoch giving the best AUROC score on the validation set, which is then used to evaluate the final performance on the test set. The protein feature encoder consists of three 1D CNN layers with the number of filters [128, 128, 128] and kernel sizes [3, 6, 9]. The drug feature encoder consists of three GCN layers with hidden dimensions [128, 128, 128]. The maximum allowed sequence length for protein is set to be 1,200, and the maximum allowed number of atoms for drug molecule is 290. In the bilinear attention module, we only employ two attention heads to provide better interpretability. The latent embedding size k is set to be 768, and the sum pooling window size s is 3. The number of hidden neurons in the fully connected decoder is 512. Our model performance is not sensitive to hyperparameter settings. The configuration details and sensitivity analysis are provided in the Supplementary Information, section 3. We also present a scalability study in the Supplementary Information, section 7.

Baselines. We compare the performance of DrugBAN with that of the following five models on DTI prediction. First and second, two shallow machine learning methods, SVM and RF, applied to the concatenated fingerprint ECFP4 and PSC features. Third, DeepConv-DTI[11], which uses CNN and one global max-pooling layer to extract local patterns in protein sequence and a fully connected network to encode drug fingerprint ECFP4. Fourth, GraphDTA[13], which models DTI using graph neural networks to encode drug molecular graphs and CNN to encode protein sequences. The learned drug and protein representation vectors are combined with a simple concatenation. To adapt GraphDTA from the original regression task to a binary classification task, we follow the steps in earlier literature[16,17] to add a Sigmoid function in its last fully connected layer, and then optimize its parameters with a cross-entropy loss. Fifth, MolTrans[17], a deep learning model that adapts transformer architecture to encode drug and protein information and uses a CNN-based interactive module to learn sub-structural interactions. For the above deep DTI models, we follow the recommended model hyperparameter settings described in their original papers.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

The experimental data used in this work are available at https://github.com/peizhenbai/DrugBAN/tree/main/datasets. All data used in this work are from public resources. The BindingDB[48] source can be found at https://www.bindingdb.org/bind/index.jsp; the BioSNAP[17,30] source can be found at https://github.com/kexinhuang12345/MolTrans/tree/master/dataset/BIOSNAP/full_data and the Human[31] source used in a previous study[16] can be found at https://github.com/lifanchen-simm/transformerCPI/blob/master/Human%2CC.elegans/dataset/human_data.txt. The co-crystalized ligands from PDB[40] are available at https://www.rcsb.org by searching their PDB IDs.

Code availability

The source code and implementation details of DrugBAN are freely available at both GitHub repository (https://github.com/peizhenbai/DrugBAN) and CodeOcean capsule (https://doi.org/10.24433/CO.3558316.v1)[57]. The code is also archived at Zenodo (https://doi.org/10.5281/zenodo.7231657)[58].

References

1. Luo, Y. et al. A network integration approach for drug-target interaction prediction and computational drug repositioning from heterogeneous information. *Nat. Commun.* **8**, 1–13 (2017).
2. Öztürk, H., Olmez, E. O. & Özgür, A. DeepDTA: deep drug-target binding affinity prediction. *Bioinformatics* **34**, i821–i829 (2018).
3. 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).
4. Zitnik, M. et al. Machine learning for integrating data in biology and medicine: principles, practice, and opportunities. *Inf. Fusion* **50**, 71–91 (2019).
5. Bagherian, M. et al. Machine learning approaches and databases for prediction of drug-target interaction: a survey paper. *Brief. Bioinform.* **22**, 247–269 (2021).
6. Wen, M. et al. Deep-learning-based drug-target interaction prediction. *J. Proteome Res.* **16**, 1401–1409 (2017).
7. Sieg, J., Flachsenberg, F. & Rarey, M. In need of bias control: evaluating chemical data for machine learning in structure-based virtual screening. *J. Chem. Inf. Model.* **59**, 947–961 (2019).
8. Lim, S. et al. A review on compound-protein interaction prediction methods: data, format, representation and model. *Comput. Struct. Biotechnol. J.* **19**, 1541–1556 (2021).
9. Gao, K. Y. et al. Interpretable drug target prediction using deep neural representation. In *Int. Joint Conf. on Artificial Intelligence* (*IJCAI*) 3371–3377 (2018).
10. Bredel, M. & Jacoby, E. Chemogenomics: an emerging strategy for rapid target and drug discovery. *Nat. Rev. Genet.* **5**, 262–275 (2004).
11. Lee, I., Keum, J. & Nam, H. DeepConv-DTI: prediction of drug-target interactions via deep learning with convolution on protein sequences. *PLoS Comput. Biol.* **15**, e1007129 (2019).

12. Hinnerichs, T. & Hoehndorf, R. DTI-Voodoo: machine learning over interaction networks and ontology-based background knowledge predicts drug-target interactions. *Bioinformatics* **37**, 4835–4843 (2021).
13. Nguyen, T. et al. GraphDTA: predicting drug-target binding affinity with graph neural networks. *Bioinformatics* **37**, 1140–1147 (2021).
14. Tsubaki, M., Tomii, K. & Sese, J. Compound protein interaction prediction with end to end learning of neural networks for graphs and sequences. *Bioinformatics* **35**, 309–318 (2019).
15. Feng, Q., Dueva, E., Cherkasov, A. & Ester, M. PADME: a deep learning-based framework for drug-target interaction prediction. Preprint at *arXiv* https://arxiv.org/abs/1807.09741 (2018).
16. Chen, L. et al. TransformerCPI: improving compound-protein interaction prediction by sequence-based deep learning with self-attention mechanism and label reversal experiments. *Bioinformatics* **36**, 4406–4414 (2020).
17. Huang, K., Xiao, C., Glass, L. & Sun, J. MolTrans: molecular interaction transformer for drug-target interaction prediction. *Bioinformatics* **37**, 830–836 (2021).
18. Schenone, M., Dancík, V., Wagner, B. K. & Clemons, P. A. Target identification and mechanism of action in chemical biology and drug discovery. *Nat. Chem. Biol.* **9**, 232–40 (2013).
19. Öztürk, H., Ozkirimli, E. & Özgür, A. WideDTA: prediction of drug-target binding affinity. Preprint at *arXiv* https://arxiv.org/abs/1902.04166 (2019).
20. Zheng, S., Li, Y., Chen, S., Xu, J. & Yang, Y. Predicting drug-protein interaction using quasi-visual question answering system. *Nat. Mach. Intell.* **2**, 134–140 (2020).
21. Abbasi, K. et al. DeepCDA: deep cross-domain compound–protein affinity prediction through lstm and convolutional neural networks. *Bioinformatics* **36**, 4633–4642 (2020).
22. Kao, P.-Y., Kao, S.-M., Huang, N.-L. & Lin, Y.-C. Toward drug-target interaction prediction via ensemble modeling and transfer learning. In *IEEE Int. Conf. on Bioinformatics and Biomedicine (BIBM)* 2384–2391 (2021).
23. Abbasi, K., Razzaghi, P., Poso, A., Ghanbari-Ara, S. & Masoudi-Nejad, A. Deep learning in drug target interaction prediction: current and future perspectives. *Curr. Med. Chem.* **28**, 2100–2113 (2021).
24. Kipf, T. & Welling, M. Semi-supervised classification with graph convolutional networks. In *Int. Conf. on Learning Representations* (ICLR, 2017).
25. Yu, Z., Yu, J., Xiang, C., Fan, J. & Tao, D. Beyond bilinear: generalized multimodal factorized high-order pooling for visual question answering. *IEEE Trans. Neural Netw. Learn. Syst.* **29**, 5947–5959 (2018).
26. Kim, J. -H., Jun, J. & Zhang, B. -T. Bilinear attention networks. In *Advances in Neural Information Processing Systems* (NeurIPS, 2018).
27. Long, M., Cao, Z., Wang, J. & Jordan, M. I. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems* (NeurIPS, 2018).
28. Weininger, D. SMILES, a chemical language and information system. 1. introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* **28**, 31–36 (1988).
29. Liu, T., Lin, Y., Wen, X., Jorissen, R. N. & Gilson, M. K. BindingDB: a web-accessible database of experimentally determined protein-ligand binding affinities. *Nucleic Acids Res.* **35**, D198–D201 (2007).
30. Zitnik, M., Sosič, R., Maheshwari, S. & Leskovec, J. BioSNAP datasets: Stanford biomedical network dataset collection. https://snap.stanford.edu/biodata (2018).
31. 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).
32. Rogers, D. & Hahn, M. Extended-connectivity fingerprints. *J. Chem. Inf. Model.* **50**, 742–754 (2010).
33. Cao, D., Xu, Q. & Liang, Y. Propy: a tool to generate various modes of chou's pseaac. *Bioinformatics* **29**, 960–962 (2013).
34. Cortes, C. & Vapnik, V. Support-vector networks. *Mach. Learn.* **20**, 273–297 (1995).
35. Ho, T. K. Random decision forests. In *Int. Conf. on Document Analysis and Recognition*, vol. 1, 278–282 (1995).
36. Ganin, Y. et al. Domain-adversarial training of neural networks. In *J. Mach. Learn. Res.* **17**, 1–35 (2016).
37. Kazokaitė, J. et al. Engineered carbonic anhydrase vi-mimic enzyme switched the structure and affinities of inhibitors. *Sci. Rep.* **9**, 1–17 (2019).
38. Rai, G. et al. Discovery and optimization of potent, cell-active pyrazole-based inhibitors of lactate dehydrogenase (ldh). *J. Med. Chem.* **60**, 9184–9204 (2017).
39. Fenalti, G. et al. Molecular control of δ-opioid receptor signalling. *Nature* **506**, 191–196 (2014).
40. Berman, H. M. et al. The protein data bank. *Nucleic Acids Res.* **28**, 235–242 (2000).
41. Jumper, J. M. et al. Highly accurate protein structure prediction with alphafold. *Nature* **596**, 583–589 (2021).
42. Pan, S. J. & Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **22**, 1345–1359 (2010).
43. Gong, B., Grauman, K. & Sha, F. Connecting the dots with landmarks: discriminatively learning domain-invariant features for unsupervised domain adaptation. In *Int. Conf. on Machine Learning* (ICML) 222-230 (2013).
44. Huang, J., Smola, A., Gretton, A., Borgwardt, K. M. & Schölkopf, B. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems* (NIPS) 601–608 (2006).
45. Li, M. et al. DGL-LifeSci: an open-source toolkit for deep learning on graphs in life science. *ACS Omega* **6**, 27233–27238 (2021).
46. Song, L., Huang, J., Smola, A. & Fukumizu, K. Hilbert space embeddings of conditional distributions with applications to dynamical systems. In *Int. Conf. on Machine Learning* (ICML) 961–968 (2009).
47. Song, L. & Dai, B. Robust low rank kernel embeddings of multivariate distributions. In *Advances in Neural Information Processing Systems* (NIPS) 3228–3236 (2013).
48. Gilson, M. K. et al. BindingDB in 2015: a public database for medicinal chemistry, computational chemistry and systems pharmacology. *Nucleic Acids Res.* **44**, D1045–D1053 (2016).
49. Bai, P. et al. Hierarchical clustering split for low-bias evaluation of drug-target interaction prediction. In *IEEE Int. Conf. on Bioinformatics and Biomedicine* (BIBM) 641–644 (2021).
50. Wishart, D. S. et al. DrugBank: a knowledgebase for drugs, drug actions and drug targets. *Nucleic Acids Res.* **36**, D901–D906 (2008).
51. Paszke, A. et al. PyTorch: an imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems* (NeurIPS, 2019).
52. Wang, M. et al. Deep graph library: a graph-centric, highly-performant package for graph neural networks. Preprint at *arXiv* https://arxiv.org/abs/1909.01315 (2019).
53. Pedregosa, F. et al. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
54. Harris, C. R. et al. Array programming with numpy. *Nature* **585**, 357–362 (2020).
55. The pandas development team. pandas-dev/pandas: Pandas 1.2.4. Zenodo https://doi.org/10.5281/zenodo.4681666 (2021).

56. Landrum, G. et al. RDKit: open-source cheminformatics. https://github.com/rdkit/rdkit (2006).
57. Bai, P., Miljković, F., John, B. & Lu, H. Interpretable bilinear attention network with domain adaptation improves drug-target prediction. CodeOcean https://doi.org/10.24433/CO.3558316.v1 (2022).
58. Bai, P., Miljković, F., John, B. & Lu, H. peizhenbai/drugban: v1.2.0. Zenodo https://doi.org/10.5281/zenodo.7231657 (2022).
59. Kim, J.-H. et al. Hadamard product for low-rank milinear pooling. In *Int. Conf. on Learning Representations* (ICLR, 2017).

Acknowledgements

We thank S. Zhou, X. Liu and L. Schöbs for their helpful suggestions and discussion on the work. P.B. is supported by a University of Sheffield Faculty of Engineering Research Scholarship (grant no. 169426530).

Author contributions

P.B., F.M., B.J. and H.L. conceived and designed the work. P.B. developed the models and performed the experiments under the guidance of B.J. and H.L. F.M. and P.B. analysed the data and conducted method comparisons. F.M. contributed to materials and the analysis tool. All authors contributed to writing the paper.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-022-00605-1.

Correspondence and requests for materials should be addressed to Haiping Lu.

Peer review information *Nature Machine Intelligence* thanks the anonymous reviewer(s) for their contribution to the peer review of this work.

nature research

Corresponding author(s): Haiping Lu

Last updated by author(s): Dec 14, 2022

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☐	☒	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☒	☐	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	☐	A description of all covariates tested
☒	☐	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☒	☐	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☒	☐	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	Python 3.8, Pandas 1.2.4 and RDKit 2021.03.2
Data analysis	Python 3.8, PyTorch 1.7.1, DGL 0.7.1, DGLlife 0.2.8, Scikit-learn 1.0.2, NumPy 1.20.2, Pandas 1.2.4, RDKit 2021.03.2 and MOE 2020.09. We also make the source code of this study available at GitHub (https://github.com/peizhenbai/DrugBAN) and Zenodo (https://doi.org/10.5281/zenodo.7231657)

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

The experimental data used in this work is available at our public repository https://github.com/peizhenbai/DrugBAN/tree/main/datasets. All datasets are from public resources. The BindingDB source is at https://www.bindingdb.org/bind/index.jsp. The BioSNAP source is at https://github.com/kexinhuang12345/MolTrans/tree/master/dataset/BIOSNAP/full_data. The Human source used in a previous study is at https://github.com/lifanchen-simm/transformerCPI/blob/master/Human%2CC.elegans/dataset/human_data.txt. The co-crystalized ligands from Protein Data Bank (PDB) are available at https://www.rcsb.org.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	We studied three public datasets: BindingDB, BioSNAP and Human, with sample sizes 49k, 27k and 6.7k respectively. These sample sizes were determined by considering the following three aspects: i) we studied the development in this area and chose sample sizes to be of the same magnitude as those in most state-of-the-art works; ii) we chose highly reputable and widely used datasets that are publicly available; iii) all interaction data used was experimentally validated. In this way, our chosen sample sizes are sufficient to facilitate a fair performance evaluation against the state-of-the-art works.
Data exclusions	After datasets were chosen as described above, no data was excluded in this work.
Replication	To verify the reproducibility of our experimental findings, we conducted five independent runs in every experiment and reported the mean and standard deviation to provide quantitative assessment of the replications. The source code and data are available at our public GitHub repository for replication by other researchers.
Randomization	We allocated data samples into experimental groups (splits) randomly with two split strategies. The first split strategy was just random split, which randomly divided drug-target pairs (data samples) into training, validation, and test sets. The second split strategy was clustering-based pair split for evaluating prediction performance on out-of-distribution data. This second strategy firstly clustered original data into clusters and then randomly split the clusters into different sets. Thus, both strategies were based on random sample allocation.
Blinding	We were blinded to the group allocation during data collection and analysis. The group allocation process was performed by computer script without any manual intervention.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☒	☐ Human research participants
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging