Genomics Proteomics Bioinformatics 17 (2019) 478–495

Genomics Proteomics Bioinformatics

www.elsevier.com/locate/gpb
www.sciencedirect.com

ORIGINAL RESEARCH

DeepCPI: A Deep Learning-based Framework for Large-scale *in silico* Drug Screening

Fangping Wan [1,#,a], Yue Zhu [2,#,b], Hailin Hu [3,#,c], Antao Dai [2,d], Xiaoqing Cai [2,e], Ligong Chen [4,f], Haipeng Gong [5,g], Tian Xia [6,h], Dehua Yang [2,*,i], Ming-Wei Wang [2,7,8,*,j], Jianyang Zeng [1,9,*,k]

[1] *Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing 100084, China*
[2] *The National Center for Drug Screening and the CAS Key Laboratory of Receptor Research, Shanghai Institute of Materia Medica, Chinese Academy of Sciences, Shanghai 201203, China*
[3] *School of Medicine, Tsinghua University, Beijing 100084, China*
[4] *School of Pharmaceutical Sciences, Tsinghua University, Beijing 100084, China*
[5] *School of Life Science, Tsinghua University, Beijing 100084, China*
[6] *Department of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan 430074, China*
[7] *School of Life Science and Technology, ShanghaiTech University, Shanghai 201210, China*
[8] *Shanghai Medical College, Fudan University, Shanghai 200032, China*
[9] *MOE Key Laboratory of Bioinformatics, Tsinghua University, Beijing 100084, China*

Received 19 March 2019; accepted 29 April 2019
Available online 6 February 2020

Handled by Yi Xing

* Corresponding authors.
E-mail: dhyang@simm.ac.cn (Yang D), mwwang@simm.ac.cn (Wang MW), zengjy321@tsinghua.edu.cn (Zeng J).
# Equal contribution.
[a] ORCID: 0000-0003-1647-3278.
[b] ORCID: 0000-0002-1106-1832.
[c] ORCID: 0000-0002-5768-4437.
[d] ORCID: 0000-0003-4241-5346.
[e] ORCID: 0000-0002-8995-4518.
[f] ORCID: 0000-0002-7893-7173.
[g] ORCID: 0000-0002-5532-1640.
[h] ORCID: 0000-0001-5984-9162.
[i] ORCID: 0000-0003-3028-3243.
[j] ORCID: 0000-0001-6550-9017.
[k] ORCID: 0000-0003-0950-7716.

Peer review under responsibility of Beijing Institute of Genomics, Chinese Academy of Sciences and Genetics Society of China.

https://doi.org/10.1016/j.gpb.2019.04.003

KEYWORDS

Deep learning;
Machine learning;
Drug discovery;
In silico drug screening;
Compound–protein interaction prediction

Abstract Accurate identification of compound–protein interactions (CPIs) *in silico* may deepen our understanding of the underlying mechanisms of drug action and thus remarkably facilitate drug discovery and development. Conventional similarity- or docking-based computational methods for predicting CPIs rarely exploit latent features from currently available large-scale unlabeled compound and protein data and often limit their usage to relatively small-scale datasets. In the present study, we propose DeepCPI, a novel general and scalable computational framework that combines effective feature embedding (a technique of representation learning) with powerful deep learning methods to accurately predict CPIs at a large scale. DeepCPI automatically learns the implicit yet expressive low-dimensional features of compounds and proteins from a massive amount of unlabeled data. Evaluations of the measured CPIs in large-scale databases, such as ChEMBL and BindingDB, as well as of the known drug–target interactions from DrugBank, demonstrated the superior predictive performance of DeepCPI. Furthermore, several interactions among small-molecule compounds and three G protein-coupled receptor targets (glucagon-like peptide-1 receptor, glucagon receptor, and vasoactive intestinal peptide receptor) predicted using DeepCPI were experimentally validated. The present study suggests that DeepCPI is a useful and powerful tool for drug discovery and repositioning. The source code of DeepCPI can be downloaded from https://github.com/FangpingWan/DeepCPI.

Introduction

Identification of compound–protein interactions (CPIs; or drug–target interactions, DTIs) is crucial for drug discovery and development and provides valuable insights into the understanding of drug actions and off-target adverse events [1,2]. Inspired by the concept of polypharmacology, *i.e.*, a single drug may interact with multiple targets [3], drug developers are actively seeking novel ways to better characterize CPIs or identify novel uses of the existing drugs (*i.e.*, drug repositioning or drug repurposing) [3,4] to markedly reduce the time and cost required for drug development [5].

Numerous computational methods have been proposed to predict potential CPIs *in silico* to narrow the large search space of possible interacting compound–protein pairs and facilitate drug discovery and development [6–12]. Although successful results can be obtained using the existing prediction approaches, several challenges remain unaddressed. First, most of the conventional prediction methods only employ a simple and direct representation of features from the labeled data (*e.g.*, established CPIs and available protein structure information) to assess similarities among compounds and proteins and infer unknown CPIs. For instance, a kernel describing the similarities among drug–protein interaction profiles [8] and the graph-based method SIMCOMP [13] were used to compare different drugs and compounds. In addition, the normalized Smith–Waterman score [9] is typically applied to assess the similarities among targets (proteins). Meanwhile, large amounts of available unlabeled data of compounds and proteins enable an implicit and useful representation of features that may effectively be used to define their similarities. Such an implicit representation of protein or compound features encoded by large-scale unlabeled data is not exploited well by most of the existing methods to predict new CPIs. Second, an increasing number of established DTIs or compound–protein-binding affinities (*e.g.*, 1 million bioassays over 2 million compounds and 10,000 protein targets in PubChem [14]) raises serious scalability issues concerning the conventional prediction methods. For instance, many similarity-based methods [7,9] require the computation of pairwise similarity scores between proteins, which is generally impractical in the setting of large-scale data. The aforementioned computational challenges highlight the need of more efficient schemes to accurately capture the hidden features of proteins and compounds from massive unlabeled data as well as the need of more advanced and scalable learning models that enable predictions from large-scale training datasets.

In machine learning communities, representation learning and deep learning (DL) are the two popular methods at present for efficiently extracting features and addressing the scalability issues in large-scale data analyses. Representation learning aims to automatically learn data representations (features) from relatively raw data that can be more effectively and easily exploited by the downstream machine learning models to improve the learning performance [15,16]. Meanwhile, DL aims to extract high-level feature abstractions from input data, typically using several layers of non-linear transformations, and is a dominant method used in numerous complex learning tasks with large-scale samples in the data science field, such as computer vision, speech recognition, natural language processing (NLP), game playing, and bioinformatics [17–19]. Although several DL models have been used to address various learning problems in drug discovery [20–22], they rarely fully exploit the currently available large-scale protein and compound data to predict CPI. For example, the computational approaches proposed in the literature [20,21] only use the hand-designed features of compounds and do not take into account the features of targets. Furthermore, these approaches generally fail to predict potential interacting compounds for a given novel target (*i.e.*, without known interacting compounds in the training data); this type of prediction is generally more urgent than the prediction of novel compounds for targets with known interacting compounds. Although a new approach—AtomNet—has been developed [23] to overcome these limitations, it can only be used to predict interacting drug partners of targets with known structures, which is often not the case in the clinical practice. In addition, despite the promising predictive performance of conventional approaches reported on benchmark datasets [24–26], few efforts have been made to

explore the extent to which these advanced learning techniques can promote the efficiency in the real drug discovery scenario.

In this article, we propose DeepCPI, a novel framework that combines unsupervised representation learning with powerful DL techniques for predicting structure-free CPIs. DeepCPI first uses the latent semantic analysis [27] and Word2vec [16,28,29] methods to learn the feature embeddings (*i.e.*, low-dimensional feature representations) of compounds and proteins in an unsupervised manner from large compound and protein corpora, respectively. Subsequently, given a compound–protein pair, the feature embeddings of both compound and protein are fed into a multimodal deep neural network (DNN) classifier to predict their interaction probability. We tested DeepCPI on several benchmark datasets, including the large-scale compound–protein affinity databases (*e.g.*, ChEMBL and BindingDB), as well as the known DTIs from DrugBank. Comparisons with several conventional methods demonstrated the superior performance of DeepCPI in numerous practical scenarios. Moreover, starting from the virtual screening initialized by DeepCPI, we identified several novel interactions of small-molecule compounds with various targets in the G protein-coupled receptor (GPCR) family, including glucagon-like peptide-1 receptor (GLP-1R), glucagon receptor (GCGR), and vasoactive intestinal peptide receptor (VIPR). Collectively, our computational test and laboratory experimentation results demonstrate that DeepCPI is a useful and powerful tool for the prediction of novel CPIs and can thus aid in drug discovery and repositioning endeavors.

Results

DeepCPI framework

The DeepCPI framework comprises two main steps (**Figure 1**): (1) representation learning for both compounds and proteins and (2) predicting CPIs (or DTIs) through a multimodal DNN. More specifically, in the first step, we use several NLP techniques to extract the useful features of compounds and proteins from the corresponding large-scale unlabeled corpora (Figures S1 and S2; Materials and methods). Here, compounds and their basic structures are regarded as "documents" and "words", respectively, whereas protein sequences and all possible three non-overlapping amino acid residues are regarded as "sentences" and "words," respectively. Subsequently, the feature embedding techniques, including latent semantic analysis [27] and Word2vec [16,28], are applied to automatically learn the implicit yet expressive low-dimensional representations (*i.e.*, vectors) of compound and protein features from the corresponding large-scale unlabeled corpora. In the second step, the derived low-dimensional feature vectors of compounds and proteins are fed into a multimodal DNN classifier to make the predictions. Further details of the individual modules of DeepCPI, including the extraction of compound and protein features, DNN model, and implementation procedure, are described in Materials and methods.

Predictive performance evaluation

We mainly evaluated DeepCPI using compound–protein pairs extracted from the currently available databases, such as ChEMBL [30] and BindingDB [31]. We first used the bioactivity data retrieved from ChEMBL [30] to assess the predictive performance of DeepCPI. Specifically, the compound–protein pairs with half maximal inhibitory concentrations (IC_{50}) or inhibition constants (K_i) ≤ 1 μM were selected as positive examples, whereas pairs with IC_{50} or $K_i \geq 30$ μM were used as negative examples. This data preprocessing step yielded 360,867 positive examples and 93,925 negative examples. To justify our criteria of selecting positive and negative examples, we mapped the known interacting drug–target pairs extracted

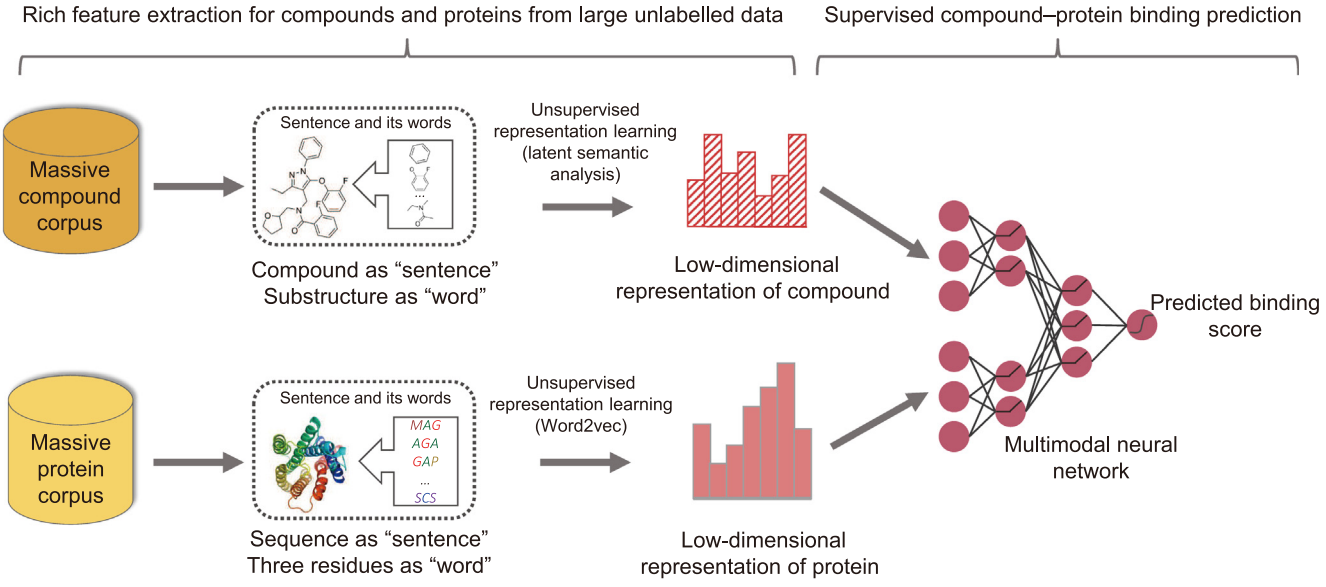

Figure 1 Schematic of the DeepCPI workflow

First, motivated by the current NLP techniques, the unsupervised representation learning strategies (including latent semantic analysis and Word2vec) are used to obtain low-dimensional representations of compound and protein features from massive unlabeled data. Subsequently, these extracted low-dimensional feature representations of compounds and proteins are fed to a multimodal DNN to make the prediction. NLP, natural language processing; DNN, deep neural network.

from DrugBank [32] (released on November 11, 2015) to the corresponding compound–protein pairs in ChEMBL (Materials and methods). The binding affinities or potencies (measured by IC_{50} or K_i) of majority of the known interacting drug–target pairs were ≤ 1 μM (>60% and 70% pairs for IC_{50} and K_i, respectively) (Figure S3). Reportedly, 1 μM is a widely-used and good indicator of strong binding affinities among compounds and proteins [33]. Therefore, we considered IC_{50} or $K_i \leq 1$ μM as a reasonable criterion for selecting positive examples. There is no well-defined dichotomy between high and low binding affinities; thus, we used a threshold of ≥ 30 μM (*i.e.*, markedly higher than 1 μM) to select negative examples, which is consistent with the method reported elsewhere [23].

To evaluate the predictive performance of DeepCPI, we considered several challenging and realistic scenarios. A computational experiment was first conducted in which we randomly selected 20% pairs from ChEMBL as the training data and the remaining pairs as the test data. This scenario mimicked a practical situation in which CPIs are relatively sparsely labeled. ChEMBL may contain similar (redundant) proteins and compounds, which may lead to over-optimistic performance resulting from easy predictions. Therefore, we minimized this effect by only retaining proteins whose sequence identity scores were < 0.40 and compounds whose chemical structure similarity scores were < 0.55 (as computed based on the Jaccard similarity between their Morgan fingerprints). More specifically, for each group of proteins or compounds with sequence identity scores ≥ 0.40 or chemical structure similarity scores ≥ 0.55, we only retained the protein or compound having the highest number of interactions and discarded the rest of the proteins or compounds in that group. The basic statistics of the ChEMBL and BindingDB datasets used in our performance evaluation are summarized in Tables S1 and S2, respectively.

Conventional cross validation, particularly leave-one-out cross validation (LOOCV), may not be an appropriate method to evaluate the performance of a CPI prediction algorithm, if the training data contain many compounds or proteins with only one interaction [34]. In such a case, training methods may learn to exploit the bias toward the proteins or compounds with a single interaction to boost the performance of LOOCV. Thus, separating the single interaction from other types of interactions during cross validation is essential [34]. Given a compound–protein interacting pair from a dataset, if the compound or protein only appeared in this interaction, we considered this pair as unique (Materials and methods). In this test scenario, we used non-unique examples as the training data and tested the predictive performance on unique pairs.

In all computational tests, three baseline methods were used for comparisons (Materials and methods). The first two were a random forest and a single-layer neural network (SLNN) with our feature extraction schemes. These were used to demonstrate the need for the DNN model. The third one was a DNN with conventional features (*i.e.*, Morgan fingerprints [35] with a radius of three for compounds and pairwise Smith–Waterman scores for proteins in the training data), which was used to demonstrate the need for our feature embedding methods. Moreover, we compared DeepCPI with two state-of-the-art network-based DTI prediction methods—DTINet [12] and NetLapRLS [10] (Materials and methods)—in a setting where redundant proteins and compounds were removed; these two methods were not used in other scenarios (Figure S4) as the cubic time and quadratic space complexities concerning the large number of compounds exceeded the limit of our server. We observed that DeepCPI significantly outperformed the network-based methods (**Figure 2**A–D). Compared to these two network-based methods, DeepCPI achieved better time and space complexities (Materials and methods), demonstrating its superiority over network-based frameworks when handling large-scale data. In addition, DeepCPI outperformed the other three baseline methods (Figure 2A–D and Figure S4) and exhibited a better prediction accuracy and generalization ability of the combination of DL and our feature extraction schemes.

Furthermore, we conducted two supplementary tests to assess the predictive performance of DeepCPI on BindingDB (Tables S1 and S2). BindingDB stores the binding affinities of proteins and drug-like small molecules [31] using the same criteria (*i.e.*, IC_{50} or $K_i \leq 1$ μM for positive examples and ≥ 30 μM for negative examples) to label compound–protein pairs. The compound–protein pairs derived from ChEMBL and BindingDB were employed as the training and test data, respectively. Compound–protein pairs from BindingDB exhibiting a compound chemical structure similarity score of ≥ 0.55 and a protein sequence identity score of ≥ 0.40 compared with any compound–protein pair from ChEMBL were regarded as overlaps and removed from the test data. The evaluation results on the BindingDB dataset demonstrated that DeepCPI outperformed all of the baseline methods (Figure 2E and F; Figure S4). Collectively, these data support the strong generalization ability of DeepCPI.

We subsequently investigated the extraction of high-level feature abstractions from the input data using the DNN. We applied T-distributed stochastic neighbor embedding (t-SNE) [36] to visualize and compare the distributions of positive and negative examples with their original 300-dimensional input features and the latent features represented by the last hidden layer in DNN. In this study, DNN was trained on ChEMBL, and a combination of 5000 positive and 5000 negative examples randomly selected from BindingDB was used as the test data. Visualization (Figure S5) showed that the test data were better organized using DNN. Consequently, the final output layer (which was simply a logistic regression classifier) can more easily exploit hidden features to yield better classification results.

Finally, we compared the performance of DeepCPI with those of the following two DL-based models: AtomNet (a structure-based DL approach for predicting compound–protein binding potencies) [23] and DeepDTI [24] (a deep belief network-based model with molecule fingerprints and protein k-mer frequencies as input features) (Materials and methods). Specifically, we compared DeepCPI with AtomNet in terms of the directory of useful decoys from DUD-E [37]. DUD-E is a widely used benchmark dataset for evaluating molecular docking programs and contains active compounds against 102 targets (Table S3). Each active compound in DUD-E is also paired with several decoys that share similar physicochemical properties but have dissimilar two-dimensional topologies, under the assumption that such dissimilarity in the compound structure results in different pharmacological activities with high probability.

We adopted two test settings as reported previously [23] to evaluate the performance of different prediction approaches

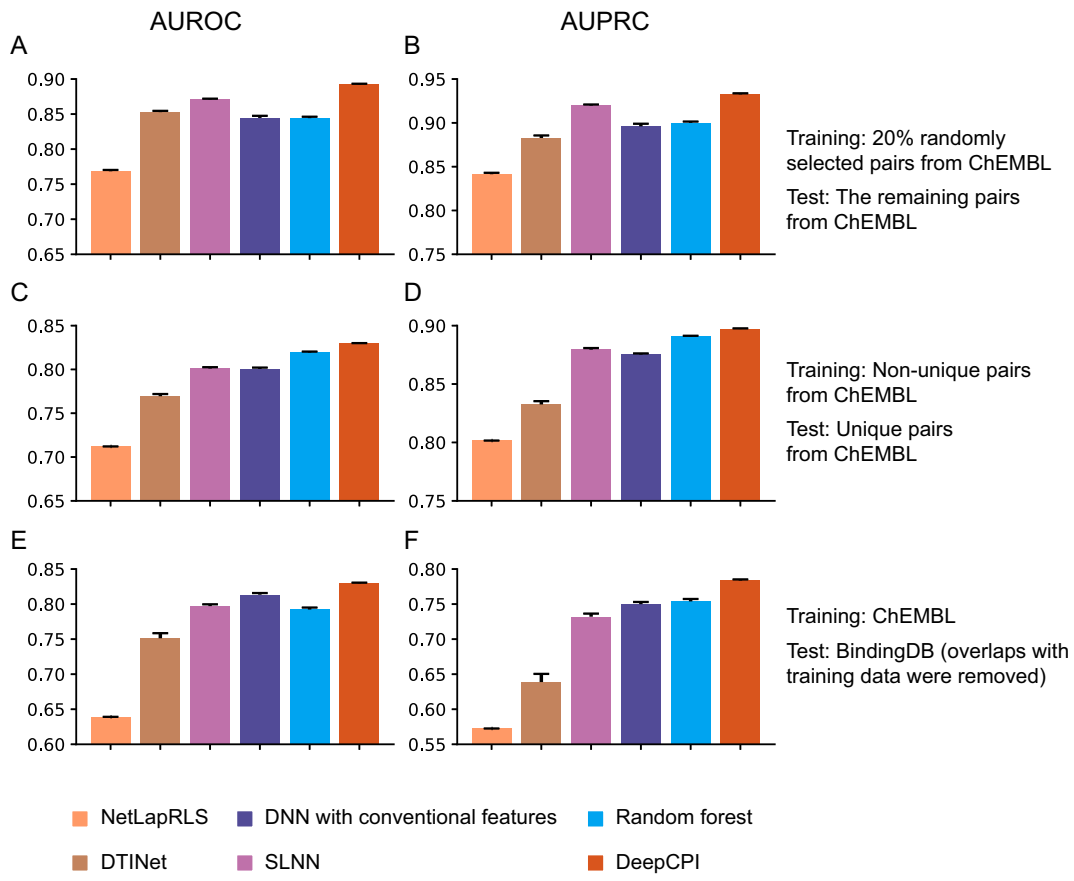

Figure 2 Performance evaluation of DeepCPI

Performance of DeepCPI was evaluated in terms of AUROC and AUPRC using different training dataset and test dataset in comparison with other state-of-the-art methods. The methods for comparison include network-based NetLapRLS and DTINet, SLNN, DNN with conventional features, and random forest. Comparison of AUROC (**A**) and AUPRC (**B**) across different methods that were trained on 20% randomly selected compound–protein pairs from ChEMBL and tested on the remaining pairs from ChEMBL. Comparison of AUROC (**C**) and AUPRC (**D**) across different methods that were trained on non-unique pairs from ChEMBL and tested on unique pairs from ChEMBL. Comparison of AUROC (**E**) and AUPRC (**F**) across different methods that were trained on ChEMBL data and tested on BindingDB data (in which the overlapping compound–protein pairs were removed). In all cases, the redundant proteins and compounds were removed. The results are summarized over 10 trials and expressed as means ± SD. SLNN, single-layer neural network; DNN, deep neural network; AUROC, area under receiver operating characteristic curve; AUPRC, area under precision-recall curve.

using DUD-E. In the first setting, cross validation was performed on 102 proteins, *i.e.*, the data were separated according to proteins. In the second setting, cross validation was performed for all pairs, *i.e.*, all compound–protein pairs were divided into three groups for validation. In addition to AtomNet, we also compared our method with a random forest model.

The tests on DUD-E showed that DeepCPI outperformed both the random forest model and AtomNet in the aforementioned settings (Table S4). In addition, in the second setting, our protein structure-free feature extraction schemes with the random forest model also greatly outperformed AtomNet, which requires protein structures and uses a convolutional neural network classifier. These observations further demonstrate the superiority of our feature extraction schemes. DeepCPI achieved a significantly larger area under receiver operating characteristic curve (AUROC) than the random forest model in the first test setting. The first setting was generally more stringent than the second one as protein information was not visible to classifiers during cross validation. Thus, this result indicates that DeepCPI has better generalization ability than the random forest model.

DeepDTI [24] requires high-dimensional features (14,564 features) as the input data; therefore, it can only be used for analyzing small-scale datasets. Thus, we mainly compared DeepCPI with DeepDTI using the 6262 DTIs provided by the original DeepDTI article [24]. We applied the same evaluation strategy as that applied in DeepDTI by randomly sampling the same number (6262) of unknown DTIs as negative examples and splitting the data into the training (60%), validation (20%), and test (20%) data. Our comparison showed that even on this small-scale dataset, DeepCPI continued to achieve a larger mean AUROC (0.9220) than DeepDTI (0.9158) (Table S5). Therefore, we believe that DeepCPI is superior to DeepDTI in terms of both predictive performance and scalability to large-scale compound affinity data.

Novel interaction prediction

All known DTI data obtained from DrugBank [32] (released on November 11, 2015) were used to train DeepCPI. The novel prediction results on the missing interactions (*i.e.*, the drug–target pairs that did not have an established interaction record in DrugBank) were then examined. Most of the top predictions with the highest scores could be supported by the evidence available in the literature. For example, among the list of the top 100 predictions, 71 novel DTIs were consistent with those reported in previous studies (Table S6). **Figure 3** presents the visualization of a DTI network comprising the top 200 predictions using DeepCPI as well as the network of the 71 aforementioned novel DTIs.

We describe several examples of these novel predictions supported by the literature below (Table S6). Specifically, in addition to the known DTIs recorded in DrugBank, DeepCPI revealed several novel interactions in neural pharmacology. These interactions may provide new direction to further decipher the complex biological processes in the treatment of neural disorders. For instance, dopamine, a catecholamine neurotransmitter with a high binding affinity for dopamine receptors (DRs), was predicted by DeepCPI to also interact with the α2 adrenergic receptor (ADRA2A). Such a prediction representing a crosstalk within the evolutionally related catecholaminergic systems is supported by the known function of dopamine acting as a weak agonist of ADRA2A [38] as well as the evidence from multiple previous animal studies [39,40]. Besides the intrinsic neurotransmitters, our prediction results involved various interesting interactions between other types of drugs and their novel binding partners. For instance, amitriptyline, a dual inhibitor of norepinephrine and serotonin reuptake, is commonly used to treat major depression and anxiety. Our predictions indicated that amitriptyline can also interact with three DR isoforms, including DRD1, DRD2, and DRD3. This result is supported by previous evidence, suggesting that amitriptyline displays binding to all three DR isoforms at sub-micromolar potencies [41].

While these antagonist potencies are relatively weak compared with those of other targets (*e.g.*, solute carrier family 6 member 2 [14] and histamine H1 receptor [42]), this new predicted interaction may offer expansion in the chemical space of antipsychotics [41] and the treatment of autism [43]. Moreover, DeepCPI predicted that oxazepam, an intermediate-acting benzodiazepine widely used in the control of alcohol withdrawal symptoms, can also act on the translocator protein, an important factor involved in intro-mitochondrial cholesterol transfer. This prediction is also supported by previous data from radioligand binding assays [44] as well as by the observation that translocator protein is responsible for the oxazepam-induced reduction of methamphetamine in rats [45].

In addition to providing novel indications in neural pharmacology, our predictions showed that polythiazide, a commonly used diuretic, can act on carbonic anhydrases. This predicted interaction, which may be related to the antihypertensive function of polythiazide [38], is supported by the evidence that polythiazide serves as a carbonic anhydrase inhibitor *in vivo* [46]. Another important branch of novel interaction predictions exemplified by an enzyme–substrate interaction between desipramine and cytochrome CYP2D6 highlighted a potential novel indication predicted by DeepCPI from a pharmacokinetics perspective. Indeed, the predicted interaction between desipramine and CYP2D6 is supported by their established metabolic association [47,48], thus offering important clinical implications in drug–drug interactions [49]. Overall, the novel DTIs predicted by DeepCPI and supported by experimental or clinical evidence in the literature further demonstrate the strong predictive performance of DeepCPI.

Validation by experimentation

As 30%–40% of the marketed drugs target GPCRs [50,51], we applied DeepCPI to identify compounds acting on this class of drug targets. In this experiment, we used positive and negative examples from ChEMBL and BindingDB as well as the compound–protein pairs with ≤ 1 μM affinities in ZINC15 [52–54] as the training data for DeepCPI. Briefly, we predicted potential interacting compounds using a dataset obtained from the Chinese National Compound Library (CNCL; http://www.cncl.org.cn/, containing 758,723 small molecules) with three class B GPCRs (GLP-1R, GCGR, and VIPR) involved in metabolic disorders and hypertension [55,56]. These proteins are challenging drug targets, particularly for the development of small-molecule modulators. For each GPCR target, we ran the trained DeepCPI model on the CNCL dataset and selected the top 100 predictions with the highest confidence scores for experimental validation as detailed below.

Pilot screening

We first conducted several pilot screening assays as an initial experimental validation step to verify the top 100 compounds that were predicted by DeepCPI to act on the aforementioned three GPCRs. For GLP-1R, a whole-cell competitive binding assay was used to examine the effects of potential positive allosteric modulators (PAMs) (**Figure 4**A; Materials and methods). For GCGR and VIPR, a cAMP accumulation assay was conducted to evaluate the agonistic and antagonistic activities of the predicted compounds (Figure 4B–E). A total of six putative ligands showed a significant augmentation of radiolabeled GLP-1 binding compared with DMSO control, *i.e.*, within the top 25% quantile of the maximum response (Figure 4A). Moreover, we discovered putative small-molecule ligands acting on GCGR and VIPR (Figure 4B–E). Among these, nine compounds exhibited significant antagonistic effects against GCGR (with 7% cAMP inhibition; Figure 4C), while one compound exhibited an obvious agonistic effect on VIPR (with 20% cAMP increase; Figure 4D). Thus, these hits were selected for further validation.

Confirmation of PAMs of GLP-1R

The six putative hits were examined for their binding to GLP-1R. Of these, three (JK0580-H009, CD3293-E005, and CD3848-F005; Figure S6) showed significant enhancement of GLP-1 binding to GLP-1R (**Figure 5**A). Their corresponding dose–response curves exhibited obvious positive allosteric effects, with half maximal effective concentration (EC_{50}) values within the low micromolar range (< 10 μM; Figure 5B). To test the specificity of the three compounds, we investigated their binding ability to GCGR, a homolog of GLP-1R. These compounds did not cross-react with GCGR (Figure 5C) but substantially promoted intracellular cAMP accumulation in

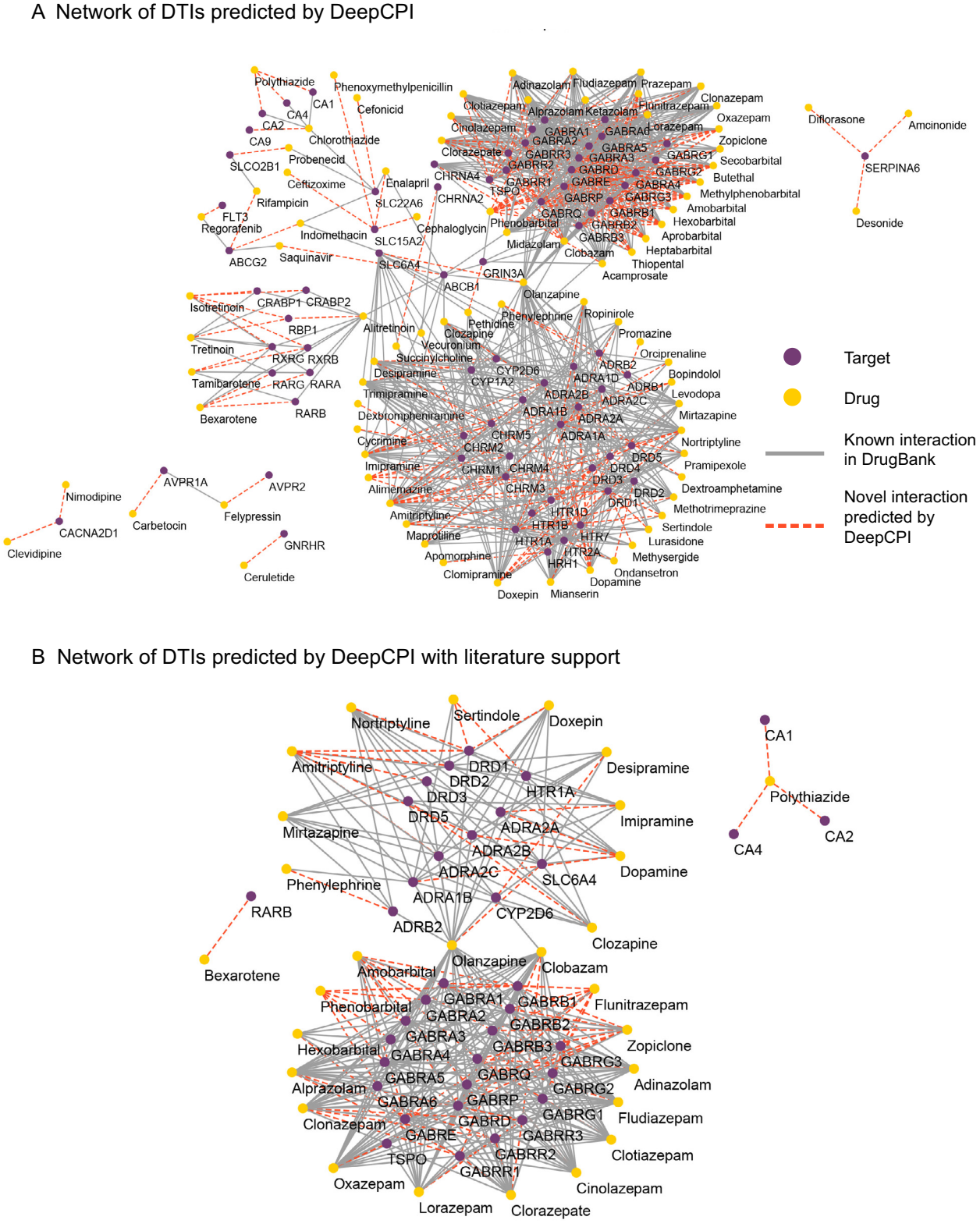

Figure 3 Network visualization of the novel DTIs predicted using DeepCPI

DeepCPI was trained on DrugBank data and used to predict novel DTIs (not recorded in DrugBank). **A.** A network comprising the top 200 novel DTI predictions. **B.** A network comprising 71 novel DTIs with literature support among the top 100 DTI predictions. Purple and yellow circles represent targets and drugs, respectively. Gray lines represent the known interactions derived from DrugBank, and red dashed lines represent the novel interactions predicted using DeepCPI. DTI, drug–target interaction.

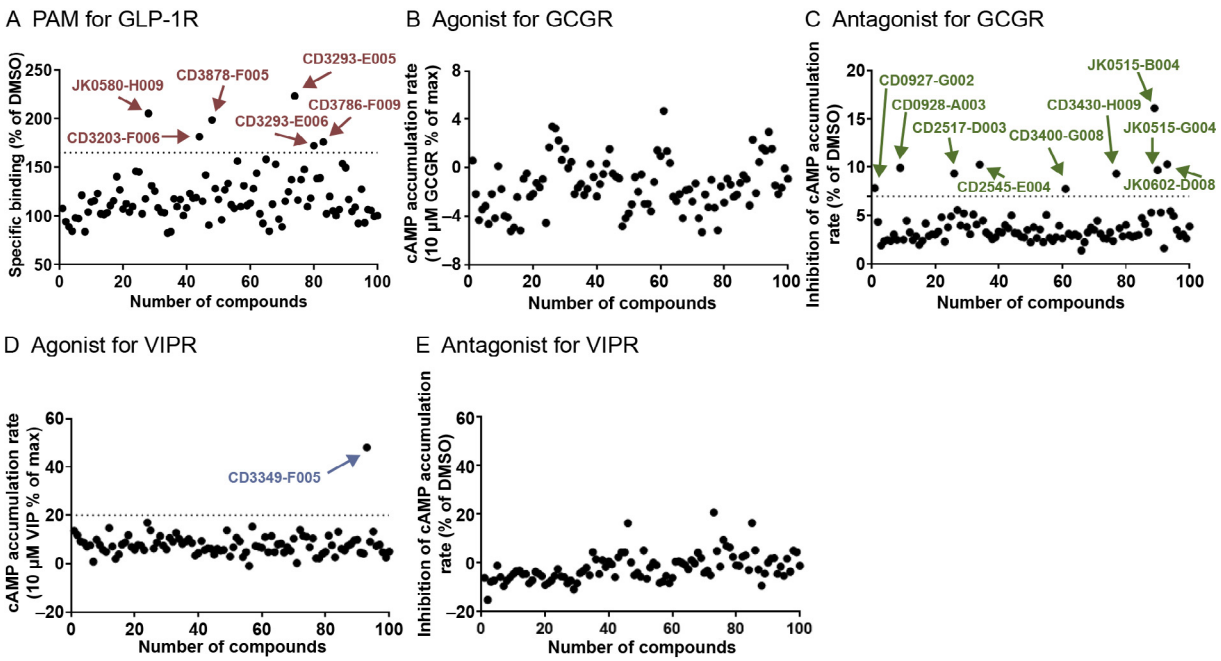

Figure 4 Pilot screening of the predicted compounds acting on GLP-1R, GCGR, and VIPR

A. A whole-cell competitive ligand binding assay was used to assess the effects of potential PAMs on GLP-1R. GLP-1R was stably expressed in FlpIn-CHO cells, and the effects of the top 100 predicted compounds (10 μM) were studied using the binding assay. Compared to DMSO, six compounds (indicated by arrows) showed obvious enhancement of radiolabeled GLP-1 binding to GLP-1R (within the top 25% quantile of the maximum response). **B.** Agonist validation for GCGR. GCGR-expressing cells were exposed to 20 μM compounds, and data were normalized with 10 nM glucagon (100%). **C.** Antagonist validation for GCGR. GCGR-expressing cells were treated with 0.01 nM glucagon and 20 μM compounds, and data were normalized to the maximal response elicited by 0.01 nM glucagon. Nine compounds (indicated by arrows) showed antagonist effects on cAMP accumulation (>7% inhibition). **D.** Agonist validation for VIPR. CHO cells transfected with VIPR were incubated with 20 μM compounds for 40 min. Data were normalized to maximal response elicited by 10 μM VIP. CD3349-F005 showed a visible agonist effect on cAMP accumulation (>20% increase). **E.** Antagonist validation for VIPR. VIP and compounds were added at the concentration of 10 nM and 20 μM, respectively, and data were normalized to maximal response elicited by 10 nM VIP. For all panels, the compounds selected for further validation are labeled with CNCL identification numbers. GLP-1R, glucagon-like peptide-1 receptor; GCGR, glucagon receptor; VIP, vasoactive intestinal peptide receptor; VIPR, VIP receptor; PAM, positive allosteric modulator; CNCL, Chinese National Compound Library.

the presence of GLP-1 (Figure 5D). Collectively, these results suggest that JK0580-H009, CD3293-E005, and CD3848-F005 are PAMs of GLP-1R.

To explore possible binding modes of these new PAMs, we conducted molecular docking studies using AutoDock Vina [57] based on the high-resolution three-dimensional active structure of GLP-1R [58] (**Figures 6** and Figure S6). We first used NNC0640 (Figure S6), a negative allosteric modulator of GLP-1R [59], as a control to demonstrate that our docking approach could recover the experimentally solved complex structure (Figure S7). Interestingly, our docking results indicated that the binding pocket for the predicted PAMs are located between transmembrane helix 5 (TM5) and TM6 of GLP-1R, which are distinct from that of NNC0640 (Figure 6) and consistent with the enlarged cavity in the active form of GLP-1R (Figure S8). Additionally, the docking results suggest that the binding sites of our predicted PAMs are located deeper inside the transmembrane domain of GLP-1R than that of the known covalently bound PAMs, including Compound 2 [59] and 4-(3-(benzyloxy)phenyl)-2-ethylsulfinyl-6-(trifluoromethyl)pyrimidine (BETP) [60,61] (Figures 6A and S7). These findings reveal a novel route for discovering and designing new PAMs of GLP-1R. To further analyze these docking results, we produced four stable cell lines expressing mutant GLP-1Rs (C347F, T149A, T355A, and I328N). As a control, we first measured the activity of BETP, which is covalently bonded to C347 in GLP-1R. Consistent with the previous findings [59], T149A mutation diminished the binding between ^{125}I-GLP-1 and GLP-1R, which could be restored by BETP treatment (**Figure 7**). Meanwhile, C347F mutation eliminated the covalent anchor of BETP and reduced its efficacy compared with that of wild-type GLP-1R (Figure 7).

However, none of the three predicted compounds exhibited binding to the T149A mutant, and their modulation behavior on C347F mutant generally aligned with that on wild-type, supporting its non-covalent binding nature (Figure 7, Table S7). These observations point to a divergent binding mode of the predicted PAMs different from that of BETP. Intriguingly, I328N mutation principally abolished the allosteric effects of the compounds (Figure 7), probably due to a large steric crash, as predicted by the docking study. In contrast, T355A mutation located at the other side of TM6 (Figure 6) showed a negligible impact on the PAM activities of the predicted compounds (Figure 7 and Table S7). Collectively, our mutagenesis results

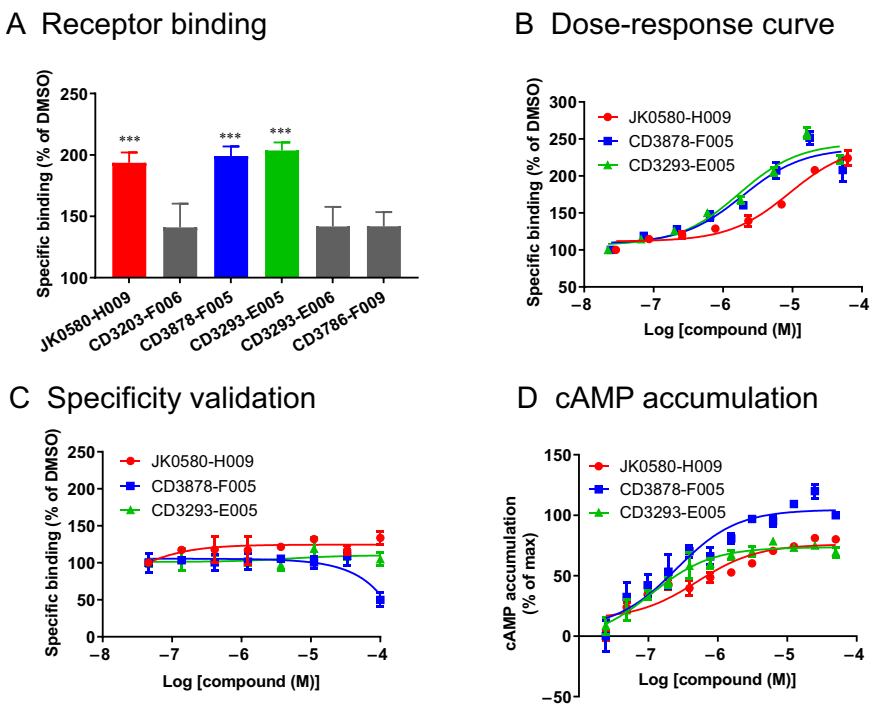

Figure 5 Experimental validation of the putative PAMs of GLP-1R

A. A receptor binding assay was used to validate the six hit compounds shown in Figure 4A. JK0580-H009, CD3878-F005, and CD3293-E005 displayed stable and significant enhancement of ^{125}I-GLP-1 binding to GLP-1R ($P < 0.0001$). **B.** Dose–response characteristics of JK0580-H009, CD3878-F005, and CD3293-E005 in terms of the binding of ^{125}I labelled GLP-1 (40 pM) to GLP-1R. **C.** Validation of specificity for compounds JK0580-H009, CD3878-F005, and CD3293-E005 using cells stably expressing GCGR. Compared to DMSO, there was no significant effect of the putative ligands on ^{125}I-glucagon binding to GCGR. **D.** Effects of compounds JK0580-H009, CD3878-F005, and CD3293-E005 on cAMP accumulation. All compounds showed dose-dependent augmentation of intracellular cAMP levels in the presence of GLP-1. All measurements were performed with at least three independent experiments, and data are shown as mean ± SEM. ***, $P < 0.0001$ (one-way ANOVA followed by Dunnett's post-test; in comparison to DMSO treatment).

support the docking data, indicating that DeepCPI can discover potential PAMs of GLP-1R.

Validation of GCGR and VIPR modulators

Nine hits with antagonistic effects against GCGR (Figure 4C) were identified in the pilot screening. Among these, CD3400-G008 (Figure S6) was confirmed to stably decrease glucagon-induced cAMP accumulation (**Figure 8**A). Subsequently, its dose dependency and estimated IC_{50} value of antagonism (22.6 μM) were determined (Figure 8B). In addition, this compound led to a rightward shift of the glucagon dose–response curve, as measured by the cAMP accumulation assay (Figure 8C). This shift corresponded to an increase in the EC_{50} value of glucagon from 23.9 pM to 5.56 pM, although it did not affect forskolin-induced cAMP accumulation (Figure 8D), ruling out the possibility that CD3400-G008 decreases cAMP accumulation in a non-specific manner. Similarly, the agonistic effect of the putative VIPR agonist CD3349-F005 (Figures 4 and S6) was dose-dependent (Figure 8E), while its agonism specificity was confirmed using a phosphodiesterase (PDE) inhibitor exclusion assay (Figure 8F). The results showed that neither 25 μM nor 50 μM of CD3349-F005 affected forskolin-induced cAMP accumulation.

Collectively, these data support the notion that DeepCPI prediction can offer a promising starting point for small-molecule drug discovery targeting GPCRs.

Discussion

In this article, we propose DeepCPI as a novel and scalable framework that combines data-driven representation learning with DL to predict novel CPIs (DTIs). By exploiting the large-scale unlabeled data of compounds and proteins, the employed representation learning schemes effectively extract low-dimensional expressive features from raw data without the requirement for information on protein structure or known interactions.

The combination of the effective feature embedding strategies and the powerful DL model is particularly useful for fully exploiting the massive amount of compound–protein binding data available from large-scale databases, such as PubChem and ChEMBL. The effectiveness of our method was fully validated using several large-scale compound–protein binding datasets as well as the known interactions between Food and Drug Administration (FDA)-approved drugs and targets. Moreover, we experimentally validated several compounds that were predicted to interact with GPCRs, which represent the largest transmembrane receptor family and probably the most important drug targets. This family constitutes >800 annotated and 150 "orphan" receptors. The latter are without known endogenous ligands and/or functions. Target-based drug discovery has been a focal point of research for decades. However, the inefficiency of mass random bioactivity

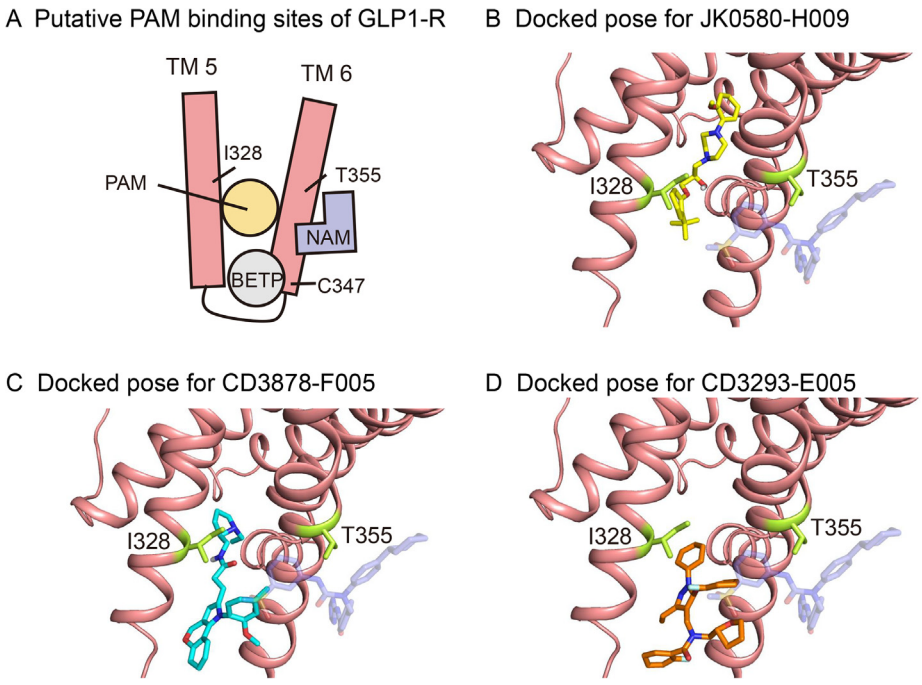

Figure 6 Molecular docking results of the predicted PAMs of GLP-1R

A. Schematic display of receptor binding sites of different PAMs including BETP and the compounds predicted by DeepCPI in the current study. T149, a key residue for GLP-1R allosteric modulation, is located on TM1 but not shown in this diagram. **B.** Docked poses for the predicted interaction between GLP-1R and JK0580-H009 (shown in yellow). **C.** Docked poses for the predicted interaction between GLP-1R and CD3878-F005 (shown in cyan) **D.** Docked poses for the predicted interaction between GLP-1R and CD3293-E005 (shown in orange). GLP-1R is shown in its active form (pink, PDB ID: 5NX2). The binding modes of the three PAMs are compared with that of NAM NNC0640 (light purple) adopted from PDB ID: 5VEX. Key residues for PAM and NAM binding, namely I328 and T355, are shown in stick and colored by lemon. TM, transmembrane; NAM, negative allosteric modulator; BETP, 4-(3-(benzyloxy)phenyl)-2-ethylsulfinyl-6-(trifluoromethyl)pyrimidine.

screening promotes the application of *in silico* prediction and discovery of small-molecule ligands. Our DeepCPI model establishes a new framework that effectively combines feature embedding with DL for the prediction of CPIs at a large scale.

We conducted several functional assays to validate our prediction results regarding the identification of small-molecule modulators targeting several class B (*i.e.*, the secretin-like family) GPCRs. GLP-1R is an established drug target for type 2 diabetes and obesity, and several peptidic therapeutic agents have been developed and marketed with combined annual sales of billions of dollars. As most therapeutic peptides require non-oral administration routes, discovery of orally available small-molecule surrogates is highly desirable. To the best of our knowledge, since the discovery of Boc5—the first non-peptidic GLP-1R agonist—more than a decade ago [62–64], little progress has been made in identifying "druggable" small-molecule mimetics for GLP-1. In this study, we identified three PAMs that were computationally predicted by DeepCPI and experimentally confirmed with bioassays to be specific to GLP-1R, thereby providing an alternative to discover non-peptidic modulators of GLP-1R.

The docking results of our predicted hits demonstrated that they could be fitted to similar sites corresponding to the binding pockets for previously known PAMs at GLP-1R in its active form. The experimental data generated by binding and cAMP accumulation assays confirmed the positive allosteric action of these hits. Overall, our modeling data, in conjunction with those obtained from mutagenesis studies, revealed the binding poses of the predicted interactions between these compounds and GLP-1R. These results offer new insights into the structural basis and underlying mechanisms of drug action.

Cross validation through different databases, supporting evidence from the known DTIs in the literature, and laboratory experimentation indicate that DeepCPI can serve as a useful tool for drug discovery and repositioning. In our follow-up studies, we intend to combine DeepCPI with additional validation experiments for the discovery of drug leads against a wide range of targets. Better prediction results may be achieved by incorporating other available data, such as gene expression and protein structures, into our DL model.

Materials and methods

DeepCPI

DeepCPI is an extension of our previously developed CPI prediction model [65]. We describe the building blocks of DeepCPI in the following three subsections.

Compound feature extraction

To learn good embeddings (*i.e.*, low-dimensional feature representations) of compounds, we used the latent semantic

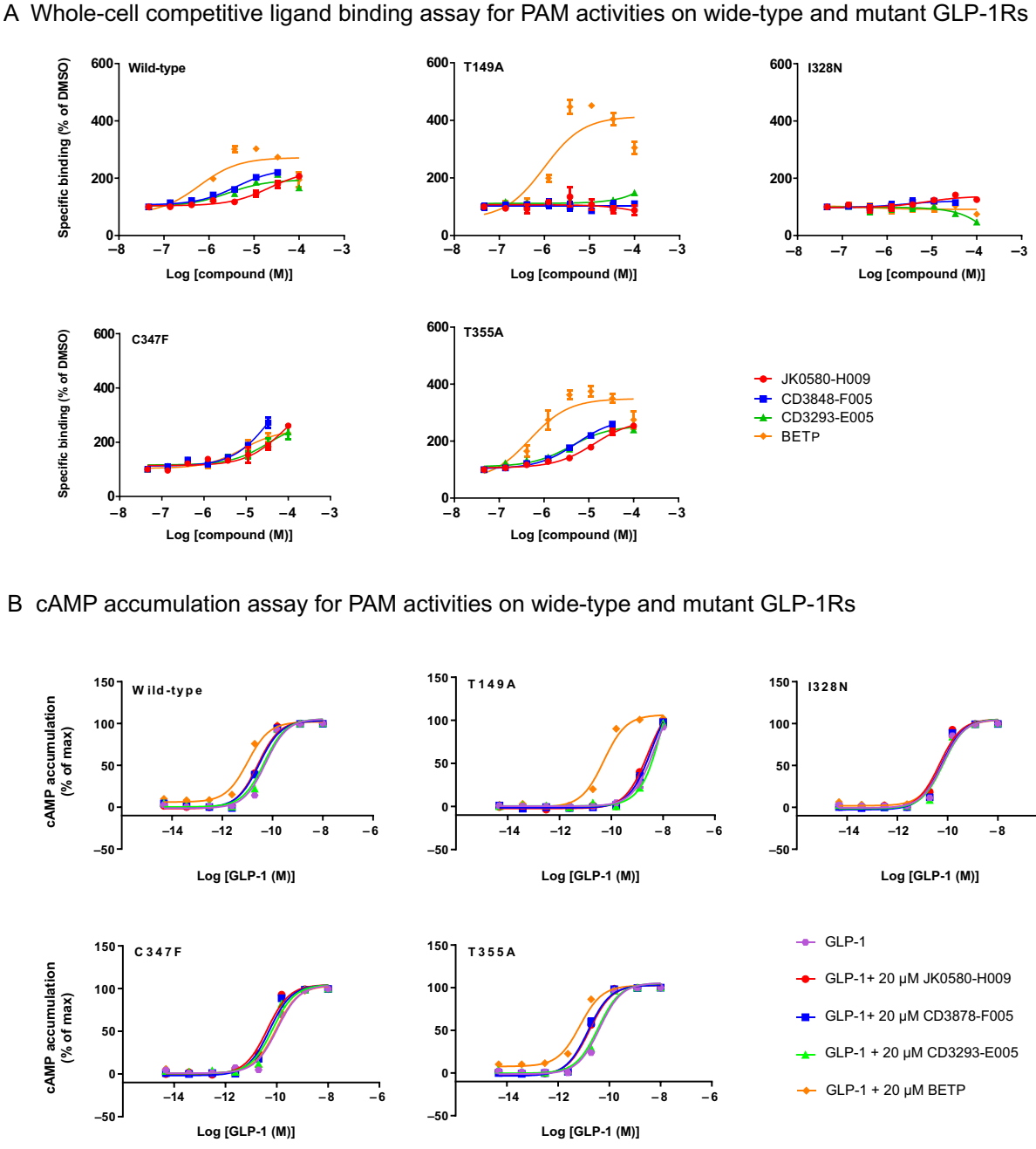

Figure 7 Mutations in GLP-1R affect the activities of predicted PAMs

According to our docking results and previous reported binding sites for allosteric modulators, we established stable cell lines expressing T149A, I328N, C347F or T355A to identify the binding pocket for predicted PAMs. **A.** PAM activities of the predicted compounds on wild-type and mutant GLP-1 receptors, measured using a whole-cell competitive ligand binding assay at different concentrations. **B.** PAM activities of the predicted compounds on the wild-type and mutant GLP-1 receptors, measured by cAMP accumulation assay. All measurements were performed with at least three independent experiments. Data are shown as means ± SEM and fitted to a four-parameter logistic regression model.

analysis (also termed latent semantic indexing) technique [27], which is probably one of the most effective methods for document similarity analyses in the field of NLP. In latent semantic analysis, each document is represented by a vector storing the term frequency or term frequency-inverse document frequency information (tf-idf). This is a numerical statistic widely used in information retrieval to describe the importance of a word in a document. Subsequently, a corpus (*i.e.*, a collection of documents) can be represented by a matrix, in which each column stores the tf-idf scores of individual terms in a document. Subsequently, singular value decomposition (SVD) is applied to obtain low-dimensional representations of features in documents.

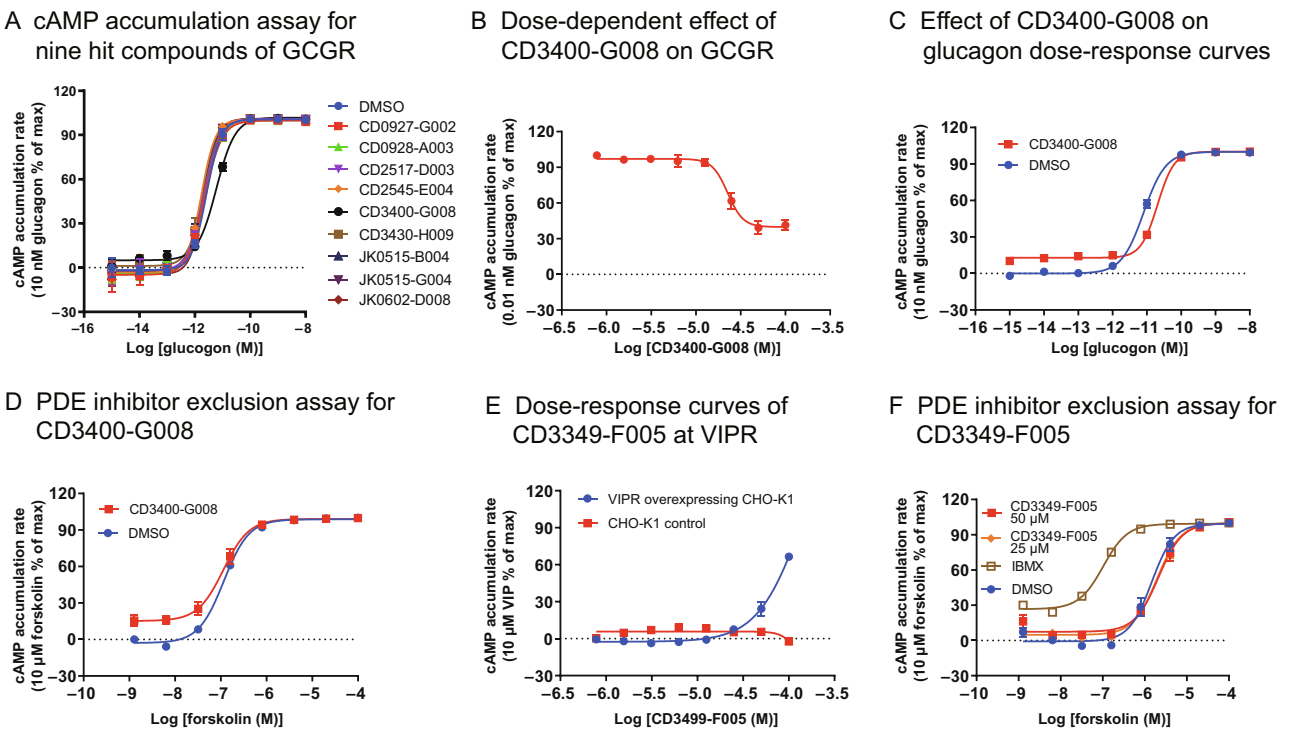

Figure 8 Experimental validation of the hit compounds acting on GCGR and VIPR

A. Validation of the nine putative modulators identified in Figure 4C using cAMP accumulation assay. CD3400-G008 displayed a stable and remarkable antagonist effect on GCGR. **B.** Dose-dependent effect of CD3400-G008 on GCGR with respect to the inhibition of cAMP accumulation. GCGR-expressing cells were incubated with 0.01 nM glucagon and treated with different concentrations of CD3400-G008 to generate the dose–response curve (IC_{50} = 22.6 μM). **C.** Glucagon dose–response curves, in which its concentration was gradually increased from 0.01 pM to 10 nM, whereas that of CD3400-G008 was set to 20 μM; DMSO was used as negative control. **D.** Effect of CD3400-G008 on forskolin-induced cAMP accumulation. **E.** Dose–response curves of CD3349-F005 in CHO-K1 cells overexpressing VIPR and in the parental CHO-K1 cells. CD3349-F005 concentration was gradually decreased from 100 μM to 0.78 μM. **F.** PDE inhibitor exclusion assay was conducted using CHO-K1 cells to detect the agonist effects of CD3349-F005 on forskolin-induced cAMP accumulation; a potent PDE inhibitor IBMX was used as positive control. All measurements were performed with at least three independent experiments in quadruplicate. Data are shown as means ± SEM and fitted to a four-parameter logistic regression model. Max, maximum response; PDE, phosphodiesterase; IBMX, 3-isobutyl-1-methylxanthine.

In the context of compound feature extraction (Figure S1), a compound and its substructures can be viewed as a document and corresponding terms, respectively. Given a compound set N, we use the Morgan fingerprints [35] with a radius of one to scan every atom of each compound in N and then generate the corresponding substructures. Let D denote the set of substructures generated from all compounds in N. We then employ a matrix $\boldsymbol{M} \in \mathbb{R}^{|D|\times|N|}$ to store the word count information for these compounds, where $\boldsymbol{M}_{ij}$ represents the tf-idf value of the i^{th} substructure in the j^{th} compound. More specifically, $\boldsymbol{M}_{ij}$ is defined as $tf(i,j)\cdot idf(i,N)$, where $tf(i,j)$ stands for the number of occurrences of the i^{th} substructure in the j^{th} compound, and $idf(i,\mathrm{N}) = log\frac{|N|}{|\{j\in D: tf(i,j)\neq 0\}|}$. Here, $\{j \in D : tf(i,j)\neq 0\}$ represents the number of documents containing the i^{th} substructure. Basically, $idf(i,N)$ reweighs $tf(i,j)$, resulting in lower weights for more common substructures and higher weights for less common substructures. This is consistent with an observation in the information theory that rarer events generally have higher entropy and are thus more informative.

After $\boldsymbol{M}$ is constructed, it is then decomposed by SVD into three matrices, $\boldsymbol{U}, \boldsymbol{\Sigma}, \boldsymbol{V}^*$, such that $\boldsymbol{M} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*$. Here, $\boldsymbol{\Sigma}$ is a $|D| \times |N|$ diagonal matrix with the eigenvalues of $\boldsymbol{M}$ on the diagonal matrix, and $\boldsymbol{U}$ is a $|D| \times |D|$ matrix in which each column $\boldsymbol{U}_i$ is an eigenvector of $\boldsymbol{M}$ corresponding to the i^{th} eigenvalue $\boldsymbol{\Sigma}_{ii}$.

To embed the compounds into a low-dimensional space $\mathbb{R}^d$, where $d < |D|$, we select the first d columns of $\boldsymbol{U}$, which correspond to the largest eigenvalues in $\boldsymbol{\Sigma}$. Let $\widehat{\boldsymbol{U}}^\top$ denote the matrix with columns corresponding to these selected eigenvectors. Subsequently, a low-dimensional embedding of $\boldsymbol{M}$ can be obtained by $\widehat{\boldsymbol{M}} = \widehat{\boldsymbol{U}}^\top\boldsymbol{M}$, where $\widehat{\boldsymbol{M}}$ is a $d \times |N|$ matrix, in which the i^{th} column corresponds to a d-dimensional embedding (or embedded feature vector) of the i^{th} compound. $\widehat{\boldsymbol{U}}^\top$ can be precomputed and fixed after being trained from a compound corpus. Given any new compound, its embedded low-dimensional feature vector can be obtained by left multiplying its tf-idf by $\widehat{\boldsymbol{U}}^\top$ (Figure S1).

Our compound feature embedding framework used the compounds retrieved from multiple sources, including all compounds labeled as active in bioassays on PubChem [14], all FDA-approved drugs in DrugBank [32], and all compounds stored in ChEMBL [30]. Duplicate compounds were removed

according to their International Chemical Identifiers (InChIs). The total number of final compounds in our compound feature extraction module used to construct matrix $\boldsymbol{M}$ was 1,795,801. The total number of different substructures generated from the Morgan fingerprints with a radius of one was 18,868. We set $d = 200$, which is a recommended value in latent semantic analysis [66].

Protein feature extraction

We applied Word2vec, a successful word-embedding technique widely used in various NLP tasks [16,28], to learn the low-dimensional representations of protein features. In particular, we use the Skip-gram with a negative sampling method [28] to train the word-embedding model and learn the context relations between words in sentences. We first introduce some necessary notations. Suppose that we are given a set of sentences S and a context window of size b. Given a sentence $s \in S$ that is represented by a sequence of words $\left(w_1^s, w_2^s, \cdots, w_L^s\right)$, where L is the length of s, the contexts of a wordw_i^s are defined as $w_{i-b}^s, \cdots, w_{i-2}^s, w_{i-1}^s, w_{i+1}^s, w_{i+2}^s, \cdots, w_{i+b}^s$. That is, all the words appearing within the context window of size b and centered at word w in the sentence are regarded as the contexts of w. We further use W to denote the set of all words appearing in S, and $\#(w, c)$ to denote the total number of occurrences of word $c \in W$ appearing in the context window of the word $w \in W$ in S. Since each word can play two roles (*i.e.*, center word and context word), the Skip-gram method equips every word $w \in W$ with two d-dimensional vector representations $\boldsymbol{e}_w, \boldsymbol{a}_w \in \mathbb{R}^d$, where $\boldsymbol{e}_w$ is used when w is a center word and $\boldsymbol{a}_w$ is used otherwise (both vectors are randomly initialized). Here, $\boldsymbol{e}_w$ or $\boldsymbol{a}_w$ basically represents the coordinates of w in the lower dimensional (*i.e.*, d-dimensional) space after embedding. Subsequently, our goal is to maximize the following objective function:

$$\sum_{w \in W} \sum_{c \in W} \#(w, c) \log \sigma(\boldsymbol{e}_w \cdot \boldsymbol{a}_w), \tag{1}$$

where $\sigma(\mathrm{x}) = \frac{1}{1+\exp(-x)}$ is the sigmoid function. Since the range of the sigmoid function is $(0, 1)$, $\sigma(\boldsymbol{e}_w \cdot \boldsymbol{a}_w)$ can be interpreted as the probability of word c being a context of word w, and Equation 1 can be viewed as the log-likelihood of a given sentence set S.

One problem in this objective function (*i.e.*, Equation 1) is that it does not take into account any negative example. If we arbitrarily assign any large positive values to $\boldsymbol{e}_w$ and $\boldsymbol{a}_w$, $\sigma(\boldsymbol{e}_w \cdot \boldsymbol{a}_w)$ would invariably be predicted as 1. In this case, although Equation 1 is maximized, such embeddings are surely useless. To tackle this problem, a Skip-gram model with negative sampling [28] has been proposed, in which "negative examples" $c_{ns}(c_{ns} \in W$ and $c_{ns} \neq w)$ are drawn from a data distribution $\mathrm{P}_{D(c_{ns})} = \frac{\#c_{ns}}{M}$, where M represents the total number of words in S and $\#c_{ns}$ represents the total number of occurrences of word c_{ns} in S. Then, the new objective function can be written as follows:

$$\begin{aligned} &\sum_{w \in W} \sum_{c \in W} \#(w, c) \log \sigma(\boldsymbol{e}_w \cdot \boldsymbol{a}_w) + k \mathbb{E}_{c_{ns} \sim \mathrm{P}_D}[\log(1 - \sigma(\boldsymbol{e}_w \cdot \boldsymbol{a}_w))] \\ &= \sum_{w \in W} \sum_{c \in W} \#(w, c) \log \sigma(\boldsymbol{e}_w \cdot \boldsymbol{a}_w) + k \mathbb{E}_{c_{ns} \sim \mathrm{P}_D}[\log(\sigma(-\boldsymbol{e}_w \cdot \boldsymbol{a}_w))], \end{aligned} \tag{2}$$

where k is the number of "negative examples" to be sampled for each observed word–context pair (w, c) during training. Maximizing this objective function can be performed using the stochastic gradient descent technique [16].

For each observed word–context pair (w, c), Equation 2 aims to maximize its log-likelihood, while minimizing the log-likelihood of k random pairs (w, c_{ns}) under the assumption that such random selections can well reflect the unobserved word–context pairs (*i.e.*, negative examples) representing the background. In other words, the goal of this task is to distinguish the observed word–context pairs from the background distribution.

As in other existing schemes for encoding the features of genomic sequences [29], each protein sequence in our framework is regarded as a "sentence" reading from its N-terminus to C-terminus and every three non-overlapping amino acid residues are viewed as a "word" (Figure S2). For each protein sequence, we start from the first, second, and third amino acid residues from the N-terminus sequentially and then consider all possible "words" while discarding those residues that cannot form a "word".

After converting protein sequences to "sentences" and all three non-overlapping amino acid residues to "words", Skip-gram with negative sampling is employed to learn the low-dimensional embeddings of these "words". Subsequently, the learnt embeddings of "words" are fixed, and an embedding of a new protein sequence is obtained by summing and averaging the embeddings of all "words" in all three possible encoded "sentences" (Figure S2). Of note, a similar approach has been successfully used to extract useful features for text classification using Word2vec [67].

In our study, the protein sequences used for learning the low-dimensional embeddings of protein features were retrieved from several databases, including PubChem [14], DrugBank [32], ChEMBL [30], Protein Data Bank [68] (www.rcsb.org), and UniProt [38]. All duplicate sequences were removed, and the final number of sequences for learning the protein features during the embedding process was 464,122. We followed the previously described principles [29] to select the hyper parameters of Skip-gram. More specifically, the embedding dimension was set to $d = 100$, the size of the context window was set to $b = 12$, and the number of negative examples was set to $k = 15$.

Multimodal DNN

Suppose that we are given a training dataset of compound–protein pairs $\{(c_i, p_i)|i = 1, 2, ..., n\}$ and a corresponding label set $\{y_i|i = 1, 2, ..., n\}$, where n stands for the total number of compound–protein pairs, $y_i = 1$ indicates that compound c_i and protein p_i interact with each other, and $y_i = 0$ otherwise. We first use the feature extraction schemes described earlier to derive the feature embeddings of individual compounds and proteins, and then feed these two embeddings to a multimodal DNN to determine whether the given compound–protein pair exhibits a true interaction.

We first introduce a vanilla DNN and then describe its multimodal variant. The basic DNN architecture comprises an input layer L_0, an output layer L_{out}, and H hidden layers L_h $(h \in \{1, 2, ..., H\})$ between input and output layers. Each

hidden layer L_h contains a set of units that can be represented by a vector $\boldsymbol{a}_h \in \mathbb{R}^{|L_h|}$, where $|L_h|$ stands for the number of units in L_h. Subsequently, each hidden layer L_h can be parameterized by a weight matrix $\boldsymbol{W}_h \in \mathbb{R}^{|L_{h-1}|\times|L_h|}$, a bias vector $\boldsymbol{b}_h \in \mathbb{R}^{|L_h|}$, and an activation function $f(\cdot)$. More specifically, the units in L_h can be calculated by $\boldsymbol{a}_h = f(\boldsymbol{W}_h\boldsymbol{a}_{h-1} + \boldsymbol{b}_h)$, where $h = 1, 2, ..., H$, and the units $\boldsymbol{a}_0$ in the input layer L_0 are the input features. We use the rectified linear unit function $f(x) = \max(0, x)$, which is a common choice of activation function in DL, perhaps due to its sparsity property, high computational efficiency, and absence of the gradient-vanishing effect during the back-propagation training process [69].

The multimodal DNN differs from the vanilla DNN in terms of the use of local hidden layers to distinguish different input modalities (Figure 1). In our case, the low-dimensional compound and protein embeddings are considered distinct input modalities and separately fed to two different local hidden layers. Subsequently, these two types of local hidden layers are concatenated and fed to joint hidden layers (Figure 1). Here, the explicit partition of local hidden layers for distinct input channels can better exploit the statistical properties of different modalities [70].

After we calculate the $\boldsymbol{a}_H$ for the final (joint) hidden layer, the output layer L_{out} is simply a logistic regression model that takes $\boldsymbol{a}_H$ as its input and computes $\hat{y} = \sigma(-\boldsymbol{w}_{out}\boldsymbol{a}_H + b_{out})$, where the output $\hat{y}$ is the confidence score for classification, σ is the sigmoid function, $\boldsymbol{w}_{out} \in \mathbb{R}^{|L_H|}$, and $b_{out} \in \mathbb{R}$ are the parameters of the output layer L_{out}. Since the sigmoid function has a range $(0, 1)$, $\hat{y}$ can also be interpreted as the interacting probability of the given compound–protein pair.

To learn $\boldsymbol{w}_{out}$, b_{out}, and all parameters $\boldsymbol{W}_h$, $\boldsymbol{b}_h$ in the hidden layers from the training data set and the corresponding label set, we need to minimize the following cross-entropy loss:

$$J = -\frac{1}{N}\sum_{i=1}^{N}[y_i log\hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i)].$$

The aforementioned minimization problem can be solved using the stochastic gradient descent and back-propagation techniques [19]. In addition, we apply two popular strategies in DL communities—dropout [71] and batch normalization [72]—to further enhance the classification performance of our DL model. In particular, dropout sets the hidden units to zero with a certain probability, which can effectively alleviate the potential overfitting problem in DL [71]. The batch normalization scheme normalizes the outputs of hidden units to zero mean and unit standard deviation, which can accelerate the training process and act as a regularizer [72].

Since positive and negative examples are possibly imbalanced, our classifier may learn a "lazy" solution. That is, in such a skewed data distribution case, the classifier can relatively easily predict the dominant class given any input. To alleviate this problem, we downsample the examples from the majority class in order for the numbers of positive and negative examples to be comparable during training. In our computational tests, we implement an ensemble version of the previously described DL model and use the average prediction over 20 models to obtain relatively more stable classification results.

Time and space complexities

Training

For compound feature extraction, given a compound, let g denote the running time of generating its Morgan fingerprints with a radius of one. The time complexity for extracting the low-dimensional representations of compound features is $O\left(|N|g + |N|^2|D|\right)$, where $|N|$ stands for the number of compounds and $|D|$ stands for the total number of substructures in all compounds. Here, $O(|N|g)$ is required for generating the substructures of all compounds and $O\left(|N|^2|D|\right)$ is required for running SVD. The space complexity for the compound feature extraction module is $O(|D| \times |N|)$. After training, we need $O(|D|d_c)$ space to store the selected eigenvectors of $\boldsymbol{M}$ for future inference, where d_c stands for the dimension of compound embedding.

For the protein feature extraction process, given a protein corpus S, it takes $O\left(|W|^2 b\right)$ time and $O\left(|W|^2\right)$ space to scan and calculate the context information, where W and b stand for the set of all possible three non-overlapping amino acid residues and the context window size, respectively. Given a word–context pair $(w, c) \in W \times W$, it takes $O\left((k + 1)d_p\right)$ time to compute the objective function in Equation 2, where d_p stands for the dimension of a protein embedding and k stands for the number of negative samples for each word–context pair. Suppose that we perform q iterations of gradient descent, then, the time complexity of the protein feature extraction module is $O\left(q|W|^2(k + 1)d_p\right)$. After training, we need $O\left(|W|d_p\right)$ space to store the protein embeddings for future inference.

For a neural network, let $|L_{h-1}|$ and $|L_h|$ denote the numbers of units in layers $h - 1$ and h, respectively. Suppose that $|L_{h-1}| \times |L_h|$ is the largest value among all layers, then, the time and space complexities for training our DL model are bounded by $O(qnH|L_{h-1}||L_h|)$ and $O(H|L_h - 1||L_h|)$, respectively, where n stands for the total number of training samples, H stands for the number of hidden layers in the neural network, and q stands for the number of training iterations.

Prediction

During the prediction stage, given a compound–protein pair, we first compute the low-dimensional vector representations of their features. This operation takes $O(g + |D|d_c)$ time for the compound and $O\left(rd_p\right)$ time for the protein, where r stands for the length of the protein sequence. Then, these two low-dimensional vector representations are fed to the deep multimodal neural network to make the prediction, which takes $O(H|L_h - 1||L_h|)$ time. In our framework, we set $|L_h - 1| = 1024$ and $|L_h| = 256$, which are small and can be considered constant (also see "Implementation of DeepCPI" below).

Mapping DrugBank data to ChEMBL

A known drug–target pair from DrugBank [32] is considered to be in ChEMBL [30] if compound–target pairs with identical InChIs for compounds or drugs and sequence identity scores $t \geq 0.40$ for proteins are present in the latter dataset. Given two protein sequences s and s', their sequence identity score t

is defined as $t = \frac{SW(s,s')}{\sqrt{SW(s,s) \times SW(s',s')}}$, where $SW(\cdot,\cdot)$ stands for the Smith–Waterman score [73].

Definition of unique compounds and proteins

Given a dataset, a compound is considered unique if its chemical structure similarity score with any other compound is < 0.55, where the chemical structure similarity score is defined as the Jaccard similarity between two Morgan fingerprints with a radius of three of the corresponding compounds [35]. Similarly, a protein is considered unique if its sequence similarity score with any other protein is < 0.40.

DeepCPI implementation

The Morgan fingerprints of compounds were generated by RDKit (https://github.com/rdkit/rdkit). Latent semantic analysis and Word2vec (Skip-gram with negative sampling) were performed using Gensim [74], a Python library designed to automatically extract semantic topics from documents. Our DNN implementation was based on Keras (https://github.com/keras-team/keras)—a highly modular DL library. For all computational experiments, we used two local hidden layers with 1024 and 256 units, respectively, for both compound and protein input channels. The two local layers were then connected to three joint hidden layers with 512, 128, and 32 units, respectively. We set the dropout rate at 0.2. Batch normalization was added to all hidden layers. During training, we selected 20% of the training data as validation set to select the optimal training epoch.

Baseline methods

When testing on ChEMBL [30] and BindingDB [31], we compared our method with two network-based DTI prediction methods, including DTINet [12] and NetLapRLS [10], as well as with three constructed baseline methods as shown below.

For DTINet and NetLapRLS, Jaccard similarity between the Morgan fingerprints of the corresponding compounds with a radius of three and pairwise Smith–Waterman scores were used to construct compound and protein similarity matrices as required in both methods. The default hyperparameters of both methods were used.

For random forest with our feature extraction schemes, we set the tree number to 128 in all computational tests as previously recommended [75]. We randomly selected 20% of the training data as validation set to select the optimal tree depth from 1 to 30.

For SLNN with our feature extraction schemes, we used a local hidden layer with 1024 units for both compound and protein input channels. The two local layers were then connected and fed to a logistic layer to make the CPI prediction. We set the dropout rate to 0.2 and batch normalization was added to the hidden layer as in DeepCPI. We selected 20% of the training data as validation set to select the optimal training epoch.

For DNN with conventional features, instead of using our feature extraction schemes, Morgan fingerprints with a radius of three and pairwise Smith–Waterman scores were used as compound and protein features, respectively. These features were subsequently fed into the same multimodal neural network as in DeepCPI. We selected 20% of the training data as validation set to select the optimal training epoch.

We also compared our method DeepCPI with two other DL-based models, namely AtomNet [23] and DeepDTI [24].

When comparing with AtomNet, we experienced difficulty in reimplementing AtomNet. Therefore, we mainly compared the performance of DeepCPI with that of AtomNet on the same DUD-E dataset. For a fair comparison, we only used a single DeepCPI model instead of an ensemble version.

We also compared the performance of DeepCPI to that of DeepDTI on 6262 DTIs provided by the original DeepDTI article [24]. DeepDTI conducted a grid search to determine the hyper parameters of the model. Hence, for a fair comparison, we followed the same strategy to determine the hyper parameters of DeepCPI. Here, we reported the hyper parameter space that we searched. In particular, we selected a batch size from $\{32, 128, 512\}$, dimensions of compound and protein features from $\{50, 60, 70, 80, 90, 100, 200, 300\}$, dropout rate from$\{0.1, 0.2\}$, and joint hidden layers with sizes from $\{\{512, 128, 32\}, \{512, 64\}\}$. We only used a single DeeCPI model instead of an ensemble version for a fair comparison.

Molecular docking

Compounds were docked using AutoDock Vina [57]. The GLP-1R model in its active form was extracted from a co-crystal structure of full-length GLP-1R and a truncated peptide agonist (PDB: 5NX2) [58]. The best docked poses were selected based on the Vina-predicted energy values.

Experimental validation

Cell culture

Stable cell lines were established using FlpIn Chinese hamster ovary (CHO) cells (Invitrogen, Carlsbad, CA) expressing either GLP-1R or GCGR and cultured in Ham's F12 nutrient medium (F12) with 10% fetal bovine serum (FBS) and 800 μg/ml hygromycin-B at 37 °C and 5% carbon dioxide (CO_2). Desired mutations were introduced to GLP-1R construct using the Muta-direct™ kit (Catalog No. SDM-15; Beijing SBS Genetech, Beijing, China) and integrated into FlpIn-CHO cells. VIPR overexpression was achieved through transient transfection using Lipofectamine 2000 (Invitrogen) in F12 medium with 10% FBS. Cells were cultured for 24 h before being seeded into microtiter plates.

Whole-cell competitive ligand binding assay

CHO cells stably expressing GLP-1R or GCGR were seeded into 96-well plates at a density of 3×10^4 cells/well and incubated overnight at 37 °C and 5% CO_2. The radioligand binding assay was performed 24 h thereafter. For homogeneous binding, cells were incubated in binding buffer with a constant concentration of ^{125}I-GLP-1 (40 pM, PerkinElmer, Boston, MA) or ^{125}I-glucagon (40 pM, PerkinElmer) and unlabeled compounds at 4 °C overnight. Cells were washed three times with ice-cold PBS and lysed using 50 μl lysis buffer (PBS supplemented with 20 mM Tris–HCl and 1% Triton X-100, pH 7.4). Subsequently, the plates were counted for radioactivity (counts per minute, CPM) in a scintillation counter (MicroBeta2 Plate Counter, PerkinElmer) using a scintillation cocktail (OptiPhase SuperMix; PerkinElmer).

cAMP accumulation assay

All cells were seeded into 384-well culture plates (4000 cells/well) and incubated for 24 h at 37 °C and 5% CO_2. For the agonist assay, after 24 h, the culture medium was discarded and 5 μl cAMP stimulation buffer [calcium- and magnesium-free Hanks' balanced salt solution (HBSS) buffer with 5 mM 4-(2-hydroxyethyl)-1-piperazineethanesulfonic acid (HEPES), 0.1% bovine serum albumin (BSA), and 0.5 mM 3-isobutyl-1-methylxanthine (IBMX)] was added to the cells. Subsequently, 5 μl compounds were introduced to simulate the cAMP reaction. For the PAM or antagonist assay, each well contained 2.5 μl cAMP stimulation buffer, 5 μl endogenous ligand (GLP-1, glucagon, or VIP) at various concentrations, and 2.5 μl testing compounds diluted in the cAMP assay buffer. After 40-min incubation at room temperature, cAMP levels were determined using the LANCE cAMP kit (Catalog No. TRF0264; PerkinElmer).

Specificity verification and PDE inhibitor exclusion assay

Two experiments were performed using the cAMP accumulation assay (antagonist mode) to study the specificity of the hit compounds acting on GCGR or VIPR. In the case of GCGR, glucagon was replaced by forskolin to investigate whether CD3400-G008 affects forskolin-induced cAMP accumulation. Forskolin concentration was gradually increased from 1.28 nM to 100 μM, whereas CD3400-G008 concentration was kept unchanged (20 μM). The PDE inhibitor exclusion assay was performed in CHO-K1 cells, in which IBMX-free stimulation buffer (calcium- and magnesium-free HBSS buffer with 5 mM HEPES and 0.1% BSA) was used. Concentrations of both IBMX (PDE inhibitor, positive control) and forskolin were gradually increased from 1.28 nM to 100 μM, and the agonistic effect of CD3349-F005 was examined at concentrations of 25 μM and 50 μM, respectively.

Availability

The source code of DeepCPI can be downloaded from https://github.com/FangpingWan/DeepCPI.

Authors' contributions

FW, DY, MWW, and JZ conceived the project. JZ, DY, and MWW supervised the project. FW and JZ designed the computational pipeline. FW implemented DeepCPI, and performed the model training and prediction validation tasks. HH and JZ analyzed the novel prediction results. YZ, AD, XC, and DY performed the experimental validation. DY and MWW analyzed the validation results. HH and HG performed the computational docking and data analysis. TX and LC helped analyze the prediction results. FW, HH, MWW, and JZ wrote the manuscript with input from all co-authors. All authors read and approved the final manuscript.

Competing interests

The authors have declared no competing interests.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 61872216 and 81630103 to JZ, 81872915 to MWW, 81573479 and 81773792 to DY), the National Science and Technology Major Project (Grant No. 2018ZX09711003-004-002 to LC), the National Science and Technology Major Project Key New Drug Creation and Manufacturing Program of China (Grant Nos. 2018ZX09735-001 to MWW, 2018ZX09711002-002-005 to DY), and Shanghai Science and Technology Development Fund (Grant Nos. 15DZ2291600 to MWW, 16ZR1407100 to AD). We acknowledge the support of the NVIDIA Corporation for the donation of the Titan X GPU used in this study.

Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.gpb.2019.04.003.

References

[1] Keiser MJ, Setola V, Irwin JJ, Laggner C, Abbas AI, Hufeisen SJ, et al. Predicting new molecular targets for known drugs. Nature 2009;462:175–81.
[2] Lounkine E, Keiser MJ, Whitebread S, Mikhailov D, Hamon J, Jenkins JL, et al. Large-scale prediction and testing of drug activity on side-effect targets. Nature 2012;486:361–7.
[3] Medina-Franco JL, Giulianotti MA, Welmaker GS, Houghten RA. Shifting from the single to the multitarget paradigm in drug discovery. Drug Discov Today 2013;18:495–501.
[4] Walsh CT, Fischbach MA. Repurposing libraries of eukaryotic protein kinase inhibitors for antibiotic discovery. Proc Natl Acad Sci U S A 2009;106:1689–90.
[5] Scannell JW, Blanckley A, Boldon H, Warrington B. Diagnosing the decline in pharmaceutical R&D efficiency. Nat Rev Drug Discov 2012;11:191–200.
[6] Keiser MJ, Roth BL, Armbruster BN, Ernsberger P, Irwin JJ, Shoichet BK. Relating protein pharmacology by ligand chemistry. Nat Biotechnol 2007;25:197–206.
[7] Martínez-Jiménez F, Marti-Renom MA. Ligand-target prediction by structural network biology using nAnnoLyze. PLoS Comput Biol 2015;11:e1004157.
[8] van Laarhoven T, Nabuurs SB, Marchiori E. Gaussian interaction profile kernels for predicting drug–target interaction. Bioinformatics 2011;27:3036–43.
[9] Bleakley K, Yamanishi Y. Supervised prediction of drug–target interactions using bipartite local models. Bioinformatics 2009;25:2397–403.
[10] Xia Z, Wu LY, Zhou X, Wong ST. Semi-supervised drug-protein interaction prediction from heterogeneous biological spaces. BMC Syst Biol 2010;4:S6.
[11] Wang Y, Zeng J. Predicting drug-target interactions using restricted Boltzmann machines. Bioinformatics 2013;29:i126–34.
[12] Luo Y, Zhao X, Zhou J, Yang J, Zhang Y, Kuang W, et al. A network integration approach for drug-target interaction prediction and computational drug repositioning from heterogeneous information. Nat Commun 2017;8:573.
[13] Hattori M, Okuno Y, Goto S, Kanehisa M. Development of a chemical structure comparison method for integrated analysis of chemical and genomic information in the metabolic pathways. J Am Chem Soc 2003;125:11853–65.

[14] Wang Y, Suzek T, Zhang J, Wang J, He S, Cheng T, et al. PubChem BioAssay: 2014 update. Nucleic Acids Res 2014;42:D1075–82.
[15] Bengio Y, Courville A, Vincent P. Representation learning: a review and new perspectives. IEEE Trans Pattern Anal Mach Intell 2013;35:1798–828.
[16] Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. arXiv 2013;1301.3781.
[17] Zhang S, Liang M, Zhou Z, Zhang C, Chen N, Chen T, et al. Elastic restricted Boltzmann machines for cancer data analysis. Quant Biol 2017;5:159–72.
[18] Hu H, Xiao A, Zhang S, Li Y, Shi X, Jiang T, et al. DeepHINT: understanding HIV-1 integration via deep learning with attention. Bioinformatics 2019;35:1660–7.
[19] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521:436–44.
[20] Unterthiner T, Mayr A, Klambauer G, Steijaert M, Wegner JK, Ceulemans H, et al. Deep learning as an opportunity in virtual screening. Workshop Deep Learn Represent Learn 2014;27:1–9.
[21] Ramsundar B, Kearnes S, Riley P, Webster D, Konerding D, Pande V. Massively multitask networks for drug discovery. arXiv 2015;1502.02072.
[22] Wan F, Hong L, Xiao A, Jiang T, Zeng J. NeoDTI: neural integration of neighbor information from a heterogeneous network for discovering new drug–target interactions. Bioinformatics 2019;35:104–11.
[23] Wallach I, Dzamba M, Heifets A. AtomNet: a deep convolutional neural network for bioactivity prediction in structure-based drug discovery. arXiv 2015;1510.02855.
[24] Wen M, Zhang Z, Niu S, Sha H, Yang R, Yun Y, et al. Deep-learning-based drug–target interaction prediction. J Proteome Res 2017;16:1401–9.
[25] Öztürk H, Özgür A, Ozkirimli E. DeepDTA: deep drug–target binding affinity prediction. Bioinformatics 2018;34:i821–9.
[26] Tsubaki M, Tomii K, Sese J. Compound–protein interaction prediction with end-to-end learning of neural networks for graphs and sequences. Bioinformatics 2019;35:309–18.
[27] Deerwester S, Dumais ST, Furnas GW, Landauer TK, Harshman R. Indexing by latent semantic analysis. J Am Soc Inf Sci 1990;41:391–407.
[28] Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J. Distributed representations of words and phrases and their compositionality. Adv Neural Inf Process Syst 2013;26:3111–9.
[29] Asgari E, Mofrad MR. Continuous distributed representation of biological sequences for deep proteomics and genomics. PLoS One 2015;10:e0141287.
[30] Bento AP, Gaulton A, Hersey A, Bellis LJ, Chambers J, Davies M, et al. The ChEMBL bioactivity database: an update. Nucleic Acids Res 2014;42:D1083–90.
[31] Liu T, Lin Y, Wen X, Jorissen RN, Gilson MK. BindingDB: a web-accessible database of experimentally determined protein-ligand binding affinities. Nucleic Acids Res 2006;35:D198–201.
[32] Wishart DS, Knox C, Guo AC, Shrivastava S, Hassanali M, Stothard P, et al. DrugBank: a comprehensive resource for in silico drug discovery and exploration. Nucleic Acids Res 2006;34:D668–72.
[33] Salvat RS, Parker AS, Choi Y, Bailey-Kellogg C, Griswold KE. Mapping the Pareto optimal design space for a functionally deimmunized biotherapeutic candidate. PLoS Comput Biol 2015;11:e1003988.
[34] van Laarhoven T, Marchiori E. Biases of drug–target interaction network data. IAPR Inter Conf Pattern Recogn Bioinformatics 2014:23–33.
[35] Rogers D, Hahn M. Extended-connectivity fingerprints. J Chem Inf Model 2010;50:742–54.
[36] Lvd Maaten, Hinton G. Visualizing data using t-SNE. J Mach Learn Res 2008;9:2579–625.
[37] Mysinger MM, Carchia M, Irwin JJ, Shoichet BK. Directory of useful decoys, enhanced (DUD-E): better ligands and decoys for better benchmarking. J Med Chem 2012;55:6582–94.
[38] UniProt Consortium. UniProt: a hub for protein information. Nucleic Acids Res 2015;43:D204–12.
[39] Cornil CA, Ball GF. Interplay among catecholamine systems: dopamine binds to α2-adrenergic receptors in birds and mammals. J Comp Neurol 2008;511:610–27.
[40] Cornil CA, Castelino CB, Ball GF. Dopamine binds to α2-adrenergic receptors in the song control system of zebra finches (*Taeniopygia guttata*). J Chem Neuroanat 2008;35:202–15.
[41] Von Coburg Y, Kottke T, Weizel L, Ligneau X, Stark H. Potential utility of histamine H3 receptor antagonist pharmacophore in antipsychotics. Bioorg Med Chem Lett 2009;19:538–42.
[42] Taylor JE, Richelson E. High affinity binding of tricyclic antidepressants to histamine H1-receptors: fact and artifact. Eur J Pharmacol 1980;67:41–6.
[43] Hellings JA, Arnold LE, Han JC. Dopamine antagonists for treatment resistance in autism spectrum disorders: review and focus on BDNF stimulators loxapine and amitriptyline. Expert Opin Pharmacother 2017;18:581–8.
[44] Schmoutz CD, Guerin GF, Goeders NE. Role of GABA-active neurosteroids in the efficacy of metyrapone against cocaine addiction. Behav Brain Res 2014;271:269–76.
[45] Spence AL, Guerin GF, Goeders NE. The differential effects of alprazolam and oxazepam on methamphetamine self-administration in rats. Drug Alcohol Depend 2016;166:209–17.
[46] Scriabine A, Korol B, Kondratas B, Yu M, P'an S, Schneider J. Pharmacological studies with polythiazide, a new diuretic and antihypertensive agent. Proc Soc Exp Biol Med 1961;107:864–72.
[47] Gueorguieva I, Jackson K, Wrighton SA, Sinha VP, Chien JY. Desipramine, substrate for CYP2D6 activity: population pharmacokinetic model and design elements of drug–drug interaction trials. Br J Clin Pharmacol 2010;70:523–36.
[48] Spina E, Gitto C, Avenoso A, Campo G, Caputi A, Perucca E. Relationship between plasma desipramine levels, CYP2D6 phenotype and clinical response to desipramine: a prospective study. Eur J Clin Pharmacol 1997;51:395–8.
[49] Reese MJ, Wurm RM, Muir KT, Generaux GT, John-Williams LS, Mcconn DJ. An in vitro mechanistic study to elucidate the desipramine/bupropion clinical drug-drug interaction. Drug Metab Dispos 2008;36:1198–201.
[50] Stevens RC, Cherezov V, Katritch V, Abagyan R, Kuhn P, Rosen H, et al. The GPCR Network: a large-scale collaboration to determine human GPCR structure and function. Nat Rev Drug Discov 2013;12:25–34.
[51] Filmore D. It's a GPCR world. Mod Drug Discovery 2004;7:24–8.
[52] Sterling T, Irwin JJ. ZINC 15–ligand discovery for everyone. J Chem Inf Model 2015;55:2324–37.
[53] Irwin JJ, Sterling T, Mysinger MM, Bolstad ES, Coleman RG. ZINC: a free tool to discover chemistry for biology. J Chem Inf Model 2012;52:1757–68.
[54] Irwin JJ, Shoichet BK. ZINC-a free database of commercially available compounds for virtual screening. J Chem Inf Model 2005;45:177–82.
[55] Roth JD, Erickson MR, Chen S, Parkes DG. GLP-1R and amylin agonism in metabolic disease: complementary mechanisms and future opportunities. Br J Pharmacol 2012;166:121–36.
[56] Munro J, Skrobot O, Sanyoura M, Kay V, Susce MT, Glaser PE, et al. Relaxin polymorphisms associated with metabolic disturbance in patients treated with antipsychotics. J Psychopharmacol 2012;26:374–9.
[57] Trott O, Olson AJ. AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. J Comput Chem 2010;31:455–61.

[58] Jazayeri A, Rappas M, Brown AJ, Kean J, Errey JC, Robertson NJ, et al. Crystal structure of the GLP-1 receptor bound to a peptide agonist. Nature 2017;546:254–8.
[59] Song G, Yang D, Wang Y, de Graaf C, Zhou Q, Jiang S, et al. Human GLP-1 receptor transmembrane domain structure in complex with allosteric modulators. Nature 2017;546:312–5.
[60] Sloop KW, Willard FS, Brenner MB, Ficorilli J, Valasek K, Showalter AD, et al. Novel small molecule glucagon-like peptide-1 receptor agonist stimulates insulin secretion in rodents and from human islets. Diabetes 2010;59:3099–107.
[61] Nolte WM, Fortin J-P, Stevens BD, Aspnes GE, Griffith DA, Hoth LR, et al. A potentiator of orthosteric ligand activity at GLP-1R acts via covalent modification. Nat Chem Biol 2014;10:629–31.
[62] Su H, He M, Li H, Liu Q, Wang J, Wang Y, et al. Boc5, a non-peptidic glucagon-like peptide-1 receptor agonist, invokes sustained glycemic control and weight loss in diabetic mice. PLoS One 2008;3:e2892.
[63] He M, Su H, Gao W, Johansson SM, Liu Q, Wu X, et al. Reversal of obesity and insulin resistance by a non-peptidic glucagon-like peptide-1 receptor agonist in diet-induced obese mice. PLoS One 2010;5:e14205.
[64] He M, Guan N, Gao WW, Liu Q, Wu XY, Ma DW, et al. A continued saga of Boc5, the first non-peptidic glucagon-like peptide-1 receptor agonist with in vivo activities. Acta Pharmacol Sin 2012;33:148–54.
[65] Wan F, Zeng J. Deep learning with feature embedding for compound-protein interaction prediction. bioRxiv 2016;086033.
[66] Bradford RB. An empirical study of required dimensionality for large-scale latent semantic indexing applications. Proc ACM Int Conf Inf Knowl Manag 2008:153–62.
[67] Iyyer M, Manjunatha V, Boyd-Graber J, Daumé III H. Deep unordered composition rivals syntactic methods for text classification. Proc Conf Assoc Comput Linguist Meet 2015;1:1681–91.
[68] Rose PW, Prlić A, Bi C, Bluhm WF, Christie CH, Dutta S, et al. The RCSB Protein Data Bank: views of structural biology for basic and applied research and education. Nucleic Acids Res 2015;43:D345–56.
[69] Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks. Proc 14th Int Conf Artif Intell Stat 2011;15:315–23.
[70] Srivastava N, Salakhutdinov RR. Multimodal learning with deep boltzmann machines. Adv Neural Inf Process Syst 2012;2:2222–30.
[71] Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 2014;15:1929–58.
[72] Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv 2015;1502.03167.
[73] Smith TF, Waterman MS. Identification of common molecular subsequences. J Mol Biol 1981;147:195–7.
[74] Rehurek R, Sojka P. Software framework for topic modelling with large corpora. Proc LREC 2010 Workshop New Challenges NLP Frameworks 2010.
[75] Oshiro TM, Perez PS, Baranauskas JA. How many trees in a random forest? Int Workshop Mach Learn Data Mining Pattern Recogn 2012:154–68.