Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category
OXFORD

Subject Section

Voodoo : Combining Bottom-up and top-down approaches through graph learning over interaction networks for drug-target-interaction prediction

Tilman Hinnerichs [1,*] and Robert Hoehndorf [2]

[1] Department, Institution, City, Post Code, Country and
[2] Department, Institution, City, Post Code, Country.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text.
Results: Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text
Availability: Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text
Contact: tilman.hinnerichs@kaust.edu.sa
Supplementary information:10264703 Supplementary data are available at *Bioinformatics* online.

1 Introduction

In history, traditional remedies, that were known for their medicinal properties lead to drugs by extraction of the functional ingredients. Alternatively, characteristics and features of potential drugs were detected by accident like in the case of penicillin. More recently, biological drug targets can be found *in silico* through discovery of suitable computational predictors.

The challenge of accurately predicting drug-target-interactions (DTI) has shown its importance in the fields of drug repurposing and repositioning, and in the exploration of novel drugs and their interaction partners. Knowledge about those links between compounds and their target proteins help in an array of medical and pharmaceutical studies. Additionally, those associations can be utilized to identify disease specific targets, leading to desirable therapeutic effects.

With the rapidly growing field of machine learning approaches and their application to bioscientifical problems in the realm of bioinformatics, different kinds of data, such as long DNA sequences could be utilized for feature generation, while rapid advances were made.

Only recently, the technique of graph learning was introduced by Kipf and Welling (2016) through graph convolution algorithms, and improved and altered under usage of different kernels (Defferrard *et al.*, 2016; Bianchi *et al.*, 2019), attention mechanisms (Veličković *et al.*, 2017), random walks (Klicpera *et al.*, 2018), and mixtures of both (Hamilton *et al.*, 2017). While based on diverse systems, they can be relevant for testing distinct hypothesis for given graphs. While convolutional filters are suitable for finding patterns among the given graph, attention mechanisms are more relevant for discovery of important regions within. Lately, graph learning approaches found application for computing compound representations for DTI prediction.

These approaches can be classified into top-down and bottom-up methods. Top-down or network approaches hereby start form the observable, phenotypical characteristics, such as side-effects, associated diseases and indications represented by knowledge graphs or ontologies, induced by a drug and infer targets based on the likely molecular mechanisms that result in these phenotypes. On the other hand, bottom-up approaches start from molecular features, such as molecular structure, molecular fingerprints, secondary structure and contact predictions and others, deriving predictions from suitable correlations between these features.

However, both bottom–up and top–down approaches to drug–target interaction prediction contain some limitations that are not solvable within themselves. Generally, it is very challenging for bottom-up methods to accurately predict, whether there a structure is binding to a protein, as not only molecular structure, but also binding sites and molecular forces are crucial for accurate prediction. Thus, finding similarities for dissimilar drugs affecting the same targets is problematic but crucial. Furthermore, phenotypic features contain relevant information for DTI prediction (Campillos *et al.*, 2008) that are complementary to knowledge gained from molecular properties. While bottom-up representations possibly contain all information needed for accurate DTI prediction, we rely on indirect information for better generalization. Contrary, top-down approaches lack the ability to spot and cope with small differences within the molecular drug structure, thus both incapable to predict on novel entities and derive similarity from structural correlation.

In order to design representations concerning both top-down and bottom-up features for proteins and drugs,respectively, we utilize protein–protein interaction (PPI) networks which have shown great results in protein function prediction (Vazquez *et al.*, 2003) and antiviral drug–target discovery (Ackerman *et al.*, 2019). However, only few applied this context to the task of DTI prediction in humans. The method of graph convolutional neural networks has been applied successfully to PPI networks (Zitnik and Leskovec, 2017), granting valuable information for medical diagnostics, closely related to phenotypic information.

Eventually, we further address potential biases and skews within DTI prediction datasets.

Introduction

Within DTI prediction, there are potential biases resulting from the underlying datasets (Pahikkala *et al.*, 2014). First, novel drugs are often designed by altering non-functional components of a drug, leading to two and more very similar drugs designed to target the same proteins (Overington *et al.*, 2006). This can result in a bias when it leads to hidden duplicates that can distribute among the train/test split, resulting in a better (measured) predictive performance than would be expected when the model is applied to identify drugs that target a protein for which no drugs yet exist. Second, some proteins (which we call *hub proteins*) have significantly more known interactions with drugs than others. In the STITCH database, 5% of the proteins have 40% of the interactions, and similar distributions are present in the Yamanishi and Drugbank (Wishart *et al.*, 2007, 2017) datasets; preferentially predicting these proteins may increase predictive performance while not reflecting the actual performance when applied to a new protein (e.g., a protein for which no interactions are known). These differences in the number of drugs targeting certain proteins may be the result of study bias where more "valuable" proteins have more drugs designed to target them due to their involvement in more common diseases (or diseases for which drugs can be more profitably marketed). This may affect common (Wang and Kurgan, 2018) cross-validation splitting schemes, that allow exploitation of these biases within DTI prediction.

Somehow smooth this transition

2 Methods

2.1 Problem Description

Voodoo aims to solve the following problem: for a given drug and a given protein we want to determine whether those interact or not. We do not differentiate between types of interaction such as activation and inhibition, and do not predict the strength of the interaction. If we additionally assume that our knowledge is complete and all drug–protein pairs without a known interaction do not interact, we can formulate the problem as a binary classification task.

2.2 Datasets

We obtain a dataset consisting of 12,884 human proteins with over 340627 links from STRING (Szklarczyk *et al.*, 2014). For the drug-target interactions, we use 229,870 links from the STITCH database (Szklarczyk *et al.*, 2015). As both STRING and STITCH provide confidence scores for each association, we filtered them as advised by a threshold of 700, therefore retaining only high-confidence interactions.

We utilize the PhenomeNET ontology (Hoehndorf *et al.*, 2011), an ontology integrating ontologies such as the Human Phenotype Ontology (Köhler *et al.*, 2018), Gene Ontology (Ashburner *et al.*, 2000; and Seth Carbon *et al.*, 2020), Mammalian Phenotype Ontology (Smith and Eppig, 2009) and several others. We obtained side effects and their links to drugs from SIDER (Kuhn *et al.*, 2015); SIDER contains side effects encoded using identifiers from the MedDRA database (Mozzicato, 2009). We mapped side effects to the PhenomeNET ontology using the *Phenomebrowser.net*, which provides a SPARQL query endpoint for the mentioned resources.

We only use proteins in our analysis that have at least one link in either STITCH or STRING, and drugs with at least one side effect and one existing target. Therefore, the intersection between these resources yields 1,428 drugs and 7,368 human proteins with 32,212 interactions for the training phase. We provide links to and methods for obtaining and processing the necessary data on Github.

For comparative evaluation, we use the gold standard dataset introduced by **?** consisting of 1,923 interactions between 708 drugs and 1,512 proteins, and BioSnap dataset (Zitnik *et al.*, 2018) which consists of 5,017 drug nodes, 2,324 gene nodes and 15,138 edges.

2.3 Model

Our model combines "top-down" and "bottom-up" information for drug–target identification. We consider an approach to be "top-down" when observable characteristics of either a drug (such as a drug effect) or protein (such as a protein function, or phenotypes resulting from a loss of function) are used to provide information about a molecular mechanisms; we consider an approach "bottom-up" when structural or other molecular information is used to determine a mechanism. In order to build a method that incorporates both top-down and bottom-up features, we first create a model for each type of feature separately. As features for the bottom-up model, we use features derived from molecular structures of drugs from the *SmilesTransformer* (Honda *et al.*, 2019) and molecular features for proteins from *DeepGOPlus* (Kulmanov and Hoehndorf, 2019). *SmilesTransformer* introduces an autoencoder, learning over the SMILES strings and therefore the molecular organization of each drug in an unsupervised manner. *DeepGOPlus* provides features derived from protein amino acid sequences which are useful to predict protein function.

As phenotypes and functions are encoded through ontologies, we use DL2Vec (Chen *et al.*, 2020) to obtain ontology based representations for use as top-down features. DL2vec constructs a graph by introducing nodes for each ontology class and edges for ontology axioms, followed by random walks starting from each node in the graph. These walks are encoded using a Word2vec (Mikolov *et al.*, 2013) model. Therefore, DL2Vec generates representations that can encode drug effects or protein functions while preserving their semantic neighbourhood within that graph.

Consider supplement, or combine with other figures.

2.3.1 Half-twin neural networks and feature transformation

As we want to learn from the similarity of drug side effects and protein phenotypes, we use a deep half-twin neural network with a contrastive loss using cosine similarity. A half-twin neural network aims to learn a similarity between two embeddings of the same dimension. The precomputed embeddings may have different dimensionality and we first process

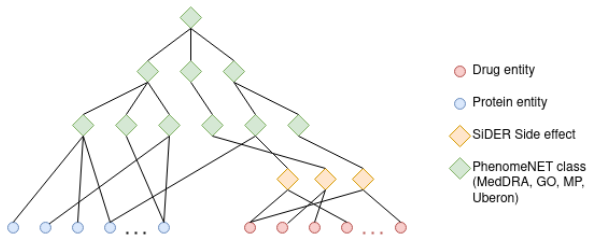

Figure 1. Drugs and proteins with annotations to SiDER and PhenomeNET

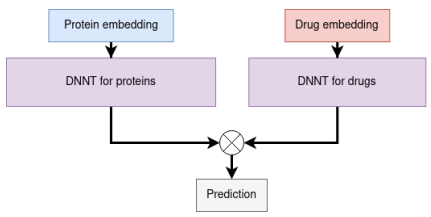

Figure 2. Half-twin network applied to molecular and DL2vec features, utilizing deep learnable feature transformations (LFT). The similarity function $\otimes$ yields the similarity between both transformed embeddings e.g. by computing the cosine similarity.

them using a learnable feature transformation (LFT) network, i.e., a fully connected neural network layer which takes as input an embedding and outputs a representation of a particular size. An example structure for both types of features can be found in Figure 2. We use this LFT to reduce the representation size of the embeddings for drugs and proteins separately. The use of an LFT enables flexible experimentation as both ontology and molecular feature for both drugs and proteins are reduced to the same dimensionality for varying sizes of inputs; this allows for a high amount of modularity across different experimental setups by adding different kinds of features into the model. Additionally, the generated features may be used for other tasks. We follow the results of *DL2vec* () and use as activation function $\sigma := \mathrm{LeakyReLU}$ which leads to improved performance compared to other activation functions.

2.3.2 Graph convolutional layers

We include these molecular and ontology-based sub-models within a graph neural network (GNN). The graph underlying the GNN is based on the protein–protein interaction (PPI) graph. The PPI dataset is represented by a graph $G = (V, E)$, where each protein is represented by a vertex $v \in V$, and each edge $e \in E \subseteq V \times V$ represents an interaction between two proteins. Additionally, we introduce a mapping $x : V \to \mathbb{R}^d$ projecting each vertex v to its node feature $x_v := x(v)$, where d denotes the dimensionality of the node features.

A graph convolutional layer (Kipf and Welling, 2016) consists of a learnable weight matrix followed by an aggregation step, formalized by

$$\mathbf{X}' = \hat{\mathbf{D}}^{-1/2}\hat{\mathbf{A}}\hat{\mathbf{D}}^{-1/2}\mathbf{X}\mathbf{\Theta} \tag{1}$$

where for a given graph $G = (V, E)$, $\hat{A} = A + I$ denotes the adjacency matrix with added self-loops for each vertex, D is described by $\hat{D}_{ii} = \sum_{j=0} \hat{A}_{ij}$, a diagonal matrix displaying the degree of each node, and Θ denotes the learnable weight matrix. Added self-loops enforce that each node representation is directly dependent on its own preceding one. The number of graph convolutional layers stacked equals the radius of relevant nodes for each vertex within the graph.

The update rule for each node is given by a message passing scheme formalized by

$$\mathbf{x}'_i = \mathbf{\Theta} \sum_j^N \frac{1}{\sqrt{\hat{d}_j \hat{d}_i}} \mathbf{x}_j \tag{2}$$

where both $\hat{d}_i, \hat{d}_j$ are dependent on the edge weights e_{ij} of the graph. With simple, single-valued edge weights such as $e_{ij} = 1 \; \forall (i, j) \in E$, all $\hat{d}_i$ reduce to d_i, i.e., the degree of each vertex i. We denote this type of graph convolutional neural layers with GCNConv.

While in this initial formulation of a GCNConv the node-wise update step is defined by the sum over all neighbouring node representations, we can alter this formulation to other message passing schemes. We can rearrange the order of activation function σ, aggregation AGG, and linear neural layer MLP with this formulation as proposed by Li *et al.* (2020a):

$$\mathbf{x}'_i = \mathrm{MLP}\left(\mathbf{x}_i + \mathrm{AGG}\left(\left\{\sigma\left(\mathbf{x}_j + \mathbf{e_{ji}}\right) + \epsilon : j \in \mathcal{N}(i)\right\}\right)\right) \tag{3}$$

where we only consider $\sigma \in \{\mathrm{ReLU}, \mathrm{LeakyReLU}\}$. We denote this generalized layer type as GENConv following the notation of PyTorch Geometric (Fey and Lenssen, 2019). While the reordering is mainly import for numerical stability, this alteration also addresses the vanishing gradient problem for deeper convolutional networks (Li *et al.*, 2020a). Additionally, we can also generalize the aggregation function to allow different weighting functions such as learnable $\mathrm{SoftMax}$ or Power for the incoming signals for each vertex, substituting the averaging step in GCNConv. Hence, while GCNConv suffers from both vanishing gradients and signal fading for large scale and highly connected graphs, each propagation step in GENConv emphasizes signals with values close to 0 and 1. The same convolutional filter and weight matrix are applied to and learned for all nodes simultaneously. We further employ another mechanism to avoid redundancy and fading signals in stacked graph convolutional networks, using residual connections and a normalization scheme (Li *et al.*, 2019, 2020b). The residual blocks are reusable and can be stacked multiple times. The structure of the GNN architecture we use is shown in Figure 3.

2.3.3 Combined prediction model

Combining half-twin and graph convolutional neural networks, we map all protein representations to their respective node features, initializing the graph convolutional update steps. The resulting representations are used for a similarity prediction as presented in figure 2. When combining ontology and molecular features with or without the graph model, we concatenate both protein features and both drugs features, before plugging them into the graph model for the similarity computation. A more in-depth image of the overall and final architecture, combining both feature types, is depicted in figure 4. Here the original representations are transformed by LFTs and then run through a stack (here with height 3) of residual graph convolutional blocks.

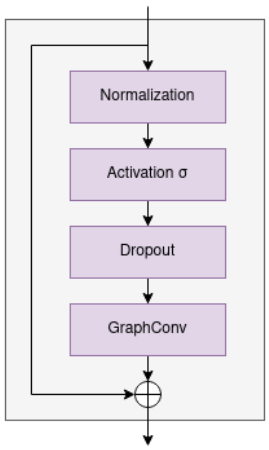

Figure 3. Residual architecture built by Li et al. (2019) and Li et al. (2020b) enabling deeper graph convolutional models

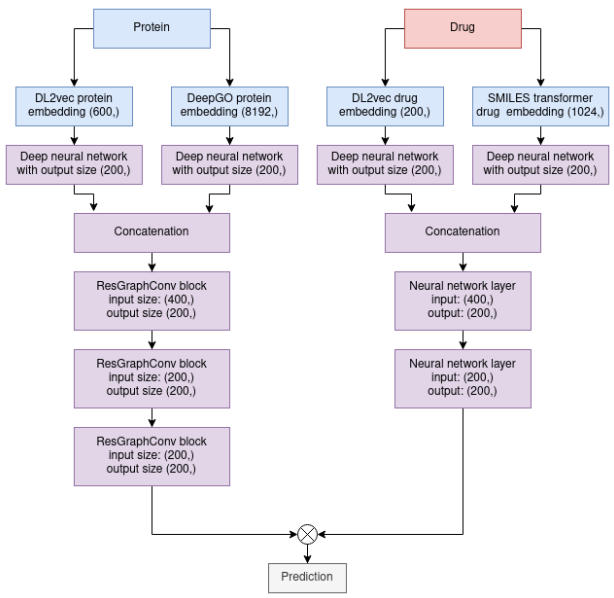

Figure 4. Residual architecture built by Li et al. (2019) and Li et al. (2020b) enabling deeper graph convolutional models

2.3.4 Hyperparameter tuning

As the number of drug-targets are sparse with respect to the number of both drugs and proteins considered, the training, validation and testing datasets are imbalanced. As there are only $22,336$ links in the considered STITCH subset, the ratio

$$w := \frac{\#drugs \cdot \#proteins}{\#dti\_links} \approx 360, \tag{4}$$

consequently needs suitable compensation in the computed loss function and appropriate metrics for the evaluation.

Therefore, we weight all positive drug-protein pair samples with this ratio by introducing the following loss function with respect to binary cross-entropy:

$$l(x, y) = -w\,[y \cdot \log x + (1 - y) \cdot \log(1 - x)] \tag{5}$$

for a given prediction x and target y, and positive weight w defined by equation (4). We average this loss among all drug-protein pairs in the training set, leading to a stable environment for the *Adam* optimization algorithm (Kingma and Ba, 2014). We implemented a 5-fold cross validation split among the proteins. Furthermore, we used early stopping in the training process.

To find the best hyperparameter configuration for the proposed model, we performed a grid search to find the most expressive and non-redundant representation. We pretrained the bottom-up and the top-down model separately and aimed at best performing models with respect to our evaluation metrics. We optimized embedding sizes, depth of the neural network, optimizer, learning rate and layer types using an extensive, manual grid search. Starting from naïve, shallow feature transformations with an embedding size of 10, we scaled the network up to residual structures with up to

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

10 hidden layers leading to embeddings of size 4000, testing different network widths and learning rates for each configuration.

2.4 Evaluation and metrics

To assess each model, we compute a variety of common metrics for binary classification. As the datasets are highly imbalanced, we use the area under the receiver operating characteristic curve (AUROC) on training, validation and testing split.

We calculate the AUROC by computing true positive rate at various false positive rate thresholds and use trapezoidal approximations to estimate the area under the curve. We refer to this measure as MacroAUC.

For RH: check this again later:

We also calculate the MicroAUC score. For given lists D and P of drugs and proteins, respectively, and a set of known interactions $Int := \{(d_i, p_i)\}$, *MicroAUC* is calculated as the average per entity (macro) *AUROC* score. Specifically, for protein-centric score, this can be formalized as: given labels $l : D \times P \to \{0, 1\}$ and predictions $y : D \times P \to [0, 1]$, we define

$$\text{MicroAUC}'_p(l, y) := \underset{p \in P}{mean} \left(\{\text{AUROC}(\{(l(d_i, p), y(d_i, p)) | d_i \in D\})\}\right)$$

In some cases, the $MicroAUC'$ score may not be defined as in some datasets some proteins or drugs have no interactions, leading to an infeasible $TPR = 0$ for all thresholds and an undefined $AUROC$ score for that entity. For those entities, we impute the $MicroAUC$ interpolating linearly, by using the accuracy for this subset:

$$\text{MicroAUC}_p(l, y) := \begin{cases} MicroAUC'_p(l, y) & \text{if } \sum_{d_i \in D} l(d_i, p) \neq 0 \\ Accuracy(l, y) & otherwise \end{cases}$$

Note that drugs and proteins can be interchanged in this formulation, and we refer to the different measures as protein-centric microAUC ($MicroAUC_p$) and a drug-centric microAUC ($MicroAUC_d$).

3 Results

3.1 Voodoo : computational model to identify drugs that target a protein

We developed Voodoo as a computational model to predict drug–target interactions. Specifically, given a protein, Voodoo will identify and rank drugs that likely target this protein. Voodoo combines two types of features: structural information for drugs and proteins that can be used to determine if the drug and protein physically interact, and information about phenotypic effects of drugs and changes in protein function that may "localize" on an interaction network (i.e., neighboring nodes will share some of these features or are phenotypically similar). As structural features, Voodoo uses structural representations of drugs from the SMILES transformer (Honda *et al.*, 2019) and representations of protein amino acid sequences from DeepGOPlus (Kulmanov and Hoehndorf, 2019). Voodoo learns representations of drug effects and protein functions using the ontology-based machine learning method DL2Vec (Chen *et al.*, 2020) and ontology-based annotations of drugs and proteins.

We construct a graph with proteins as nodes and protein-protein interactions as edges, mapping the protein features to each target as node features. Voodoo then propagates information among the PPI network utilizing graph convolutional steps, calculates the similarity of drug and protein representations, and predicts whether there is an interaction. The full workflow scheme is depicted in figure 5

We evaluate our model's ability to identify drug–target interactions using different approaches and datasets. First, we perform a cross-validation over proteins and validate our results. A cross-validation over proteins aims to evaluate how the model performs when tasked to identify drugs that may target a "novel" protein, i.e., one not seen during training, or a protein for which a drug that targets it should be predicted.

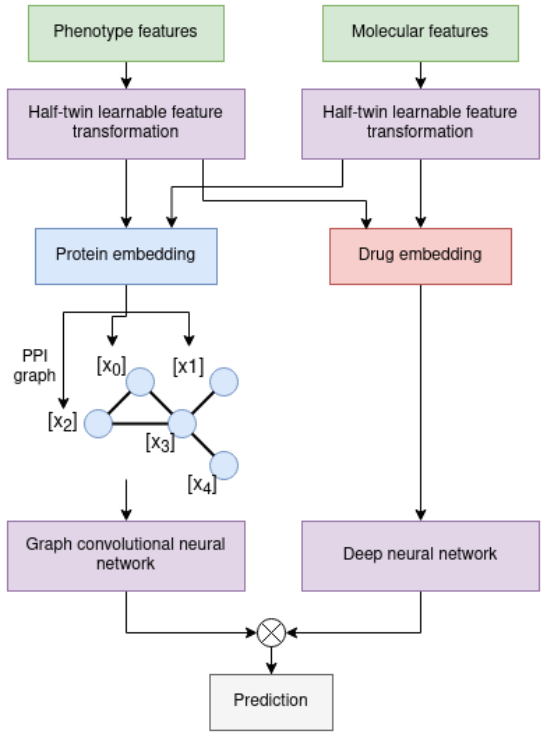

Figure 5. Full DTI prediction model based on the pretrained learnable feature transformations (LFT) for either molecular structure or ontology based features. The transformed protein representations are added to each corresponded protein as node features for the graph convolutional steps.

We trained, validated and finally tested all considered models on the STITCH dataset using a 5-fold cross-validation over a protein split; we then selected the best-performing models (with respect to $MicroAUC_p$, see Section 2.4) and evaluate them in a 5-fold protein-split cross-validation on the Yamanishi benchmark dataset to avoid validation overfitting and yield more realistic testing results. To evaluate the influence of the different features separately, and to determine whether they "localize" on the PPI graph (and therefore can be exploited successfully by the graph neural networks), we train and evaluate models with different types of features, and with and without inclusion of the PPI graph, separately. We comparing the molecular (MolPred) and phenotype-based (OntoPred) prediction model, and a combination of both where we concatenate both types of features. Table 1 shows the results of these experiments.

We find that the model using ontology-based features (*OntoPred*) is showing better performance on STITCH compared to using only molecular features. We also observe that only the model using ontology-based features results in increased performance when incorporating the PPI graph. This increase can be observed with different graph neural network architectures and configurations. While the *GCNConv* and *GENConv* architecture already shows some minor improvement, the use of *ResGraphConv* results in larger performance improvements. *ResGraphConv* blocks add a large amount of additional learnable parameters to the network, leading to more expressive power. To test whether the observed improvement is due to the number of learnable parameters added or the result of better exploiting the information about PPIs, we experiment with a graph model

check – not currently in the table

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

	(a) STITCH results				(b) Yamanishi results			
Voodoo results	PPI graph				PPI graph			
	without		with		without		with	
	Macro AUC	Micro AUC_p	Macro AUC	Micro AUC_p	Macro AUC	Micro AUC	Macro AUC	Micro AUC_p
MolPred	0.69	0.65	0.69	0.67	0.66	0.67	0.66	0.64
OntoPred	0.88	0.87	0.92	0.93	0.80	0.79	0.83	0.82
Voodoo (MolPred + OntoPred)	0.89	0.90	**0.93**	**0.94**	0.83	0.82	**0.84**	**0.84**

Approach	Original Splitting scheme	(c) Yamanishi results			(d) BioSnap results		
		Original scheme	Protein split		Original scheme	Protein split	
		Macro AUC	Macro AUC	Micro AUC_p	Macro AUC	Macro AUC	Micro AUC_p
Naive predictor	DP pairs	0.85	–	–	0.79	–	–
DTINet	DP pairs	0.91	0.74	0.67	–	–	–
DTIGEMS+	DP pairs	**0.93**	0.72	0.68	–	–	–
DTI-CDF	Proteins	0.85	**0.85**	0.79	–	–	–
DeepDTI	Drugs	–	–	–	0.88	0.76	0.70
DeepDTA	DP pairs	–	–	–	0.88	0.77	0.69
DeepConv-DTI	DP pairs	–	–	–	0.88	0.76	0.73
MolTrans	DP pairs	–	–	–	0.90	0.77	0.74
Voodoo	Proteins	0.84	0.84	**0.84**			

Table 1. (a) + (b) Results for Voodoo on STITCH and Yamanishi dataset evaluated with a 5-fold cross-validation. We hereby denote the molecular feature based predictor with MolPred, while abbreviating the ontology based, top-down predictor with OntoPred. (c) + (d) Results for various state of the art (c) drug–target interaction prediction methods on Yamanishi dataset and (d) drug–target affinity prediction methods on BIOSNAP dataset, evaluated on their original and the protein cross-validation splitting scheme, approximately reproducing the results of MolTrans CITATION.

in which all graph convolutional neural layers in the residual blocks are removed, resulting in a model with similar parameters but without the ability to use graph-based information. This pruned network, with no information on the protein–protein interactions, reached very similar results to the original *OntoPred* model and showed no improvement.

The improvements when including the graph are only provided by the *GENConv* graph convolution scheme which includes the *ResGraphConv* blocks; *GCNConv* and other graph convolutional methods fail to achieve any gain in comparison to the plain *OntoPred* performance even when combined with the residual blocks. The discrepancy between *GENConv* and other graph convolutional methods may be the result of numerical instability and fading signals ().

Our results demonstrate that the inclusion of graph information can increase performance when ontology-based features are used but not when molecular features are used alone. This observation allows us to conclude that information about protein functions localizes on the graph whereas molecular features do not.

3.2 Protein-centric evaluation

The goal of Voodoo is to find candidate drugs that target a specific protein; however, so far, we do not evaluate this application but rather how Voodoo would perform in finding plausible drug–target interactions among all possible interactions (since we use the MacroAUC as our main evaluation measure). This evaluation does not correspond to the application of Voodoo in finding drugs that target a specific protein. To provide a better estimate on how Voodoo performs for individual protein targets, we use micro-averages between proteins and compute the *MicroAUC* (see Section 2.4); to determine *MicroAUC*, we average the performance (true and false positive rates) per protein instead of across all drug–protein pairs; the resulting measure can therefore better estimate how Voodoo performs when tasked with finding a drug that targets a specific protein.

Furthermore, we hypothesize that it may be possible to exploit biases in drug–target interactions to achieve relatively high prediction performance without obtaining a biologically meaningful signal. For example, hub proteins may have a large number of interactions, or certain drugs interact with many proteins, and preferentially predicting these interactions may increase predictive performance even in the absence of any biological features. To test this hypothesis, we design a "naïve" baseline model that predicts the same list of proteins for each drug based only on the number of known drug–target interactions for a protein. Formally, given lists D and P of drugs and proteins and a set of known interactions $\mathcal{I} := \{(d_i, p_i)\}$, we construct an interaction matrix $M_{int} \in \{0,1\}^{|D|\times|P|}$ with

$$M_{ij} = \begin{cases} 1 & \text{if } (d_i, p_j) \in \mathcal{I} \\ 0 & otherwise \end{cases}$$

describing for all drug–protein pairs whether there is a known interaction or not. We now rank all proteins $p_j \in P$ descending by their number of drug interactors by summing over the columns of M_{ij} and ranking these sums:

$$f : P \to \mathbb{N} \text{ with } f : p_j \mapsto \sum_{i=1}^{|D|} M_{ij}$$

Our "naïve" predictor P_k predicts all drugs to interact with the top k targets with respect to the introduced ranking:

$$P_k : D \times P \to \{0,1\} \text{ with } P_k : d_i, p_j \mapsto \begin{cases} 1 & \text{if } p_j \in Top_k(P) \\ 0 & otherwise \end{cases}$$

with the only hyperparameter k.

The prediction P_k is not dependent on the drug d_i and will predict the same ranked list of drugs for all proteins; consequently, this naïve predictor does not rely on any biological features and will not predict any

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

novel information about interactions between drugs and proteins; the naïve predictor only exploits imbalances in the evaluation set to make predictions that may perform well.

The way in which we formulated the naïve predictor, it is not applicable for a protein split cross-validation as the number of interactions for each protein in the validation set is unknown. We apply this naive predictor on both the STITCH and Yamanishi datasets, using the full datasets as well as a 5-fold cross-validation over drugs and over drug–protein pairs to compare the prediction results directly to Voodoo . For each fold, we gradually increase k to determine the best performance for each fold. Using the full dataset, drug–target split, and drug split, we obtain the following MacroAUC results

Put in Results table, then remove from here

: for the STITCH database, we obtain a performance of 0.76 on the whole dataset, 0.70 for the drug–target pairs and 0.73 in case of the drug splitting scheme; on the Yamanishi dataset, we obtain MacroAUC scores of 0.88, 0.84 and 0.85, for the total dataset, drug–target pair and drug split, respectively. The naïve predictor shows higher performance on the Yamanishi dataset than on STITCH, and a substantial gain in comparison to an expected random predictor on both datasets. In the following, we utilize this naïve predictor as baseline to compare its performance to state of the art models and Voodoo .

For comparison with the state of the art methods, we chose the best performing methods for drug–target interaction prediction that were previously evaluated on the Yamanishi benchmark dataset. These methods include DTIGEMS+ (Thafar *et al.*, 2020) and DTI-CDF (Chu *et al.*, 2019) which have showed superior results in comparison to numerous works. Furthermore, we added DTINet (Luo *et al.*, 2017) as method for comparison which has been used to develop a number of methods such as NeoDTI (Wan *et al.*, 2019) with similar methodology.

We evaluate all models on their recommended splitting scheme choice, hyperparameters and folds in cross-validation, measuring their respective AUROC. We further evaluate each model by performing a protein-wise cross-validation determining the MacroAUC and $MicroAUC_p$. For this evaluation, we allow sub-sampling of negatives for the training process but not for the validation and testing phase as real world applications of these models would have to deal with possibly imbalanced data.

Check after restructure

The results of our experiments are summarized in part (c) of Table 1; we calculated the performance of all compared methods over their original splitting scheme and over a protein split. We find that there is a large difference in performance when evaluating over a drug–target pair split compared to a protein split, with generally higher performance achieved when using the drug–target pair split. Second, when evaluating the same methods over a protein split, we find a substantial performance difference in comparison to the splitting scheme used in the original evaluation of each method. DTI-CDF was originally evaluated on all three splitting schemes underlining this point (). While Voodoo provides comparable performance to the naïve predictor and DTI-CDF in terms of MacroAUC, it yields considerably better results with respect to MicroAUC$_p$. We also find that methods that are trained using a protein split generally result in higher MicroAUC$_p$ than methods trained using a drug–target pair split.

As the difference in performance with different splitting schemes is quite large, we further evaluated additional drug–target interaction and drug–target affinity prediction methods that were trained and evaluation on other datasets. Following the results of MolTrans (Huang *et al.*, 2020), we reevaluated DeepDTI (Wen *et al.*, 2017), DeepDTA (Öztürk *et al.*, 2018), DeepConv-DTI (Lee *et al.*, 2019) and MolTrans itself on BioSnap () over a protein split, as shown in part (d) of Table 1. The authors of MolTrans evaluated all considered methods over the drug–target pair and the protein split; we were able to reproduce the results (Table 1 (d)),

What is that? Is it described in methods?

check table

check table

showing a substantial difference based on the splitting scheme. We additionally computed the $MicroAUC_p$ score for all considered methods, leading to evaluation results that are comparable to the ones obtained on the Yaminishi dataset.

are the datasets comparable?

add Voodoo model to table to allow DIRECT comparison:

The results of the analysis are summarized in the upper part of Figure 1.

We find that...

summarize key findings from Table.

Replace/merge table:

4 Discussion

4.1 “Bottom-up” and “top-down” prediction of drug-target interactions

There are many computational methods to predict drug–target interactions. They can broadly be grouped in two types; the first, which we refer to as “bottom-up” approaches, start from molecular information about a drug and protein and predict an interaction based on their molecular properties; the second, which we refer to as “top-down” approaches, start from observable characteristics of an organism and infer drug–target interactions as the putative molecular mechanisms that explain these observations; many top-down approaches follow methods similar to those used in reverse genetics ().

Another view on these two approaches is as direct and indirect ways to predict drug–target interactions. Molecular information can be used to directly determine whether two molecules (such as a drug and protein) have the ability to interact, whereas information about phenotypic consequences of a drug (drug effects) or disruption of a protein function can be used to indirectly suggest candidate drug–target interactions. Molecular information will be specific to a drug–target pair and we would not expect this information to propagate through a protein–protein interaction network; the main information about drug–target interactions that could be obtained from interactions between proteins is information about binding sites between proteins that may also be used by a drug molecule (i.e., information that protein P_1 binds to protein P_2 reveals information about the molecular structures of both P_1 and P_2). On the other hand, phenotypic consequences of changes in protein function, or drug effects, are often a result of aberrant pathway or network activity and involve more than one protein; consequently, we expect these features to benefit more from including information about protein–protein interactions. Our results confirm this hypothesis and demonstrate that molecular features do not benefit from including the interaction network whereas the indirect, top-down features benefit from the propagation over the interaction network.

There are other types of indirect features that could be added to our model. A common feature that is added are drug indications which are highly predictive of drug–target interactions (); however, we do not include them in our model as including drug indications would allow our model to make many trivial predictions based only on remembering which targets are often used for which indication; including network information would likely benefit predictions based on drug indications because different drugs may target the same pathway through different mechanisms ().

we built protein function and ontology based features based on DL2vec

Ontology derived protein function focused features are highly predictive for dtis

We built a versatile template for various features to test localization on the PPI graph

picture(

(-1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

normal GCNs don't work on PPI graph, as it is highly connected → needs stronger more expressive aggregation function → GENConv in residual blocks for better numerical stability

molecular features work for all (most?) drugs, top-down features only for those that have a side-effect profile

- Voodoo comparison to other approaches on BioSnap prone to bias, as $BioSnap : drugs \times prots = 5500 \times 3000$, while only ≈ 1000 drugs have Sider Annotation, and

4.2 Evaluating drug–target interaction predictions

We spent a significant amount of time testing different evaluation schemes for predicting drug–target interactions. We designed a naïve predictor and find that it can achieve a performance only slightly below state of the art drug–target prediction methods. We investigated this finding further and

In the presence of imbalanced

Only few other methods perform their split over proteins (Wang and Kurgan, 2018), DTI-CDF does it

Running split over proteins is harder than, drug and drug protein pair split (see below table)

this applies for both DTI prediction and drug target affinity prediction (and Saras gene-disease association)

Stratified Cross validation is suitable for training, but **NOT** for validating and testing (Uselessly high AUPRC)

microAUC is a superior and more intuitive metric for drug repurposing → why for each protein and not for each drug

Likely discussion:

The aim of Voodoo is to predict candidate drugs that target a given protein; the challenge is to develop a training and evaluation scheme that does not simply overfit to the inherent biases in training and testing data. In general, when performing cross-validation for DTI prediction, the options are to split over

1. split over drugs,
2. split over drug–target pairs, or
3. split over proteins

where the first and third option concern splitting drugs and proteins, respectively, into train, validation and test sets, and arranging the corresponding drug-target interactions. They ensure that at least parts of the interactions are not seen during training and evaluate either how well targets are predicted for unseen drugs or unseen proteins. Hereby, different training and prediction schemes lead to divergent expressiveness of the resulting model.

Likely discussion:

The most common scheme for DTI prediction is the split over drug-target pairs (Wang and Kurgan, 2018), where likely all drugs and targets of the validation and testing phase have already occurred in the training phase, as part of other drug-target pairs. The second most prevalent arrangement is the split over drugs, while only close to none is aiming on a protein split. However, the first and second splitting scheme are exposed to the first dataset bias and are hence more likely vulnerable to transductive inference by just predicting recently seen structures, rather than implementing inductive inference and generalizing over the drug representations. Second, these two strategies are more susceptible to the second bias, as only in these cases the model may overfit on the number of existing interactions for a single protein, while in the third scheme the number of interactions of the test proteins is entirely unknown during training process.

Likely discussion:

Assuming a hypothetical, perfectly generalizing model built upon an unbiased dataset, this very model would yield similar performances for all three , not overfitting on the known structures. On the other hand, for a hypothetical entirely overfitting model, trained on a highly biased dataset, this model would show substantial deviations from the original performance over another split.

Likely discussion:

We emphasize, that all real-world models are prone to some sort of overfitting, and unknown, deviant entities in both validation and testing set will likely lead to some sort of performance gap for relevant metrics. However, a large disparity may hint the biases stated above.

A naive predictor (ranking proteins) and predict each drug similarly achieves cutting edge performance (87.5 AUROC for whole dataset, 85.5 for 5-fold cross validation in drug-split) → No prot focused microAUC possible. → hub proteins

Yamanishi Dataset is only partially suitable for comparing results, if everybody just derives a suitable subset (DTIGEMS)

This also applies to drug target affinity prediction. We were hereby able to roughly reproduce the results from MolTrans (Bioinformatics) on BioSnap

Discussion:

The lack of stratification, only impacts the not considered area under precision recall curve (AUPRC) and not the Macro AUROC score, while also supporting the expressiveness of $MicroAUC_p$ with more data points.

5 Conclusions

For the bottom-up approach we build a model that only relies on molecular features, which we will discuss in more detail in the following methods chapter. For the combination of both approaches we now attach the predictions to the protein-protein interaction graph as node features for future graph learning steps. In this graph we tried to find both patterns and regions for each drug that could be of interest through application of different graph convolutional layers, which in return represent the feature for each protein. Representing the drug we take the drug-drug interaction graph and the semantic similarity over side effects which we will explain in the following paragraphs.

Combining bottom-up and top-down features improves state of the art.

Evaluating DTI predictions is non-trivial due to numerous biases in the training and evaluation datasets.

6 Conclusion

Acknowledgements

Funding

This work has been supported by the... Text Text Text Text.

References

Ackerman, E. E., Alcorn, J. F., Hase, T., and Shoemaker, J. E. (2019). A dual controllability analysis of influenza virus-host protein-protein interaction networks for antiviral drug target discovery. *BMC Bioinformatics*, **20**(1).

and Seth Carbon, Douglass, E., Good, B. M., Unni, D. R., Harris, N. L., Mungall, C. J., Basu, S., Chisholm, R. L., Dodson, R. J., Hartline, E., Fey, P., Thomas, P. D., Albou, L.-P., Ebert, D., Kesling, M. J., Mi, H., Muruganujan, A., Huang, X., Mushayahama, T., LaBonte, S. A., Siegele, D. A., Antonazzo, G., Attrill, H., Brown, N. H., Garapati, P., Marygold, S. J., Trovisco, V., dos Santos, G., Falls, K., Tabone, C., Zhou, P., Goodman, J. L., Strelets, V. B., Thurmond, J., Garmiri, P., Ishtiaq, R., Rodríguez-López, M., Acencio, M. L., Kuiper, M., Lægreid, A., Logie, C., Lovering, R. C., Kramarz, B., Saverimuttu, S. C. C., Pinheiro, S. M., Gunn, H., Su, R., Thurlow, K. E., Chibucos, M., Giglio, M., Nadendla, S., Munro, J., Jackson, R., Duesbury, M. J., Del-Toro, N., Meldal, B. H. M., Paneerselvam, K., Perfetto, L., Porras, P., Orchard, S., Shrivastava, A., Chang, H.-Y., Finn, R. D., Mitchell, A. L., Rawlings, N. D., Richardson, L., Sangrador-Vegas, A., Blake, J. A., Christie, K. R., Dolan, M. E., Drabkin, H. J., Hill, D. P., Ni, L., Sitnikov, D. M., Harris, M. A., Oliver, S. G., Rutherford, K., Wood, V., Hayles, J., Bähler, J., Bolton, E. R., Pons, J. L. D., Dwinell, M. R., Hayman, G. T., Kaldunski, M. L., Kwitek, A. E., Laulederkind, S. J. F., Plasterer, C., Tutaj, M. A., Vedi, M., Wang,

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

S.-J., D'Eustachio, P., Matthews, L., Balhoff, J. P., Aleksander, S. A., Alexander, M. J., Cherry, J. M., Engel, S. R., Gondwe, F., Karra, K., Miyasato, S. R., Nash, R. S., Simison, M., Skrzypek, M. S., Weng, S., Wong, E. D., Feuermann, M., Gaudet, P., Morgat, A., Bakker, E., Berardini, T. Z., Reiser, L., Subramaniam, S., Huala, E., Arighi, C. N., Auchincloss, A., Axelsen, K., Argoud-Puy, G., Bateman, A., Blatter, M.-C., Boutet, E., Bowler, E., Breuza, L., Bridge, A., Britto, R., Bye-A-Jee, H., Casas, C. C., Coudert, E., Denny, P., Estreicher, A., Famiglietti, M. L., Georghiou, G., Gos, A., Gruaz-Gumowski, N., Hatton-Ellis, E., Hulo, C., Ignatchenko, A., Jungo, F., Laiho, K., Mercier, P. L., Lieberherr, D., Lock, A., Lussi, Y., MacDougall, A., Magrane, M., Martin, M. J., Masson, P., Natale, D. A., Hyka-Nouspikel, N., Orchard, S., Pedruzzi, I., Pourcel, L., Poux, S., Pundir, S., Rivoire, C., Speretta, E., Sundaram, S., Tyagi, N., Warner, K., Zaru, R., Wu, C. H., Diehl, A. D., Chan, J. N., Grove, C., Lee, R. Y. N., Muller, H.-M., Raciti, D., Auken, K. V., Sternberg, P. W., Berriman, M., Paulini, M., Howe, K., Gao, S., Wright, A., Stein, L., Howe, D. G., Toro, S., Westerfield, M., Jaiswal, P., Cooper, L., and Elser, J. (2020). The gene ontology resource: enriching a GOld mine. *Nucleic Acids Research*, **49**(D1), D325–D334.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. (2000). Gene ontology: tool for the unification of biology. *Nature Genetics*, **25**(1), 25–29.

Bianchi, F. M., Grattarola, D., Alippi, C., and Livi, L. (2019). Graph neural networks with convolutional arma filters.

Campillos, M., Kuhn, M., Gavin, A.-C., Jensen, L. J., and Bork, P. (2008). Drug target identification using side-effect similarity. *Science*, **321**(5886), 263–266.

Chen, J., Althagafi, A., and Hoehndorf, R. (2020). Predicting candidate genes from phenotypes, functions and anatomical site of expression. *Bioinformatics*.

Chu, Y., Kaushik, A. C., Wang, X., Wang, W., Zhang, Y., Shan, X., Salahub, D. R., Xiong, Y., and Wei, D.-Q. (2019). DTI-CDF: a cascade deep forest model towards the prediction of drug-target interactions based on hybrid features. *Briefings in Bioinformatics*, **22**(1), 451–462.

Defferrard, M., Bresson, X., and Vandergheynst, P. (2016). Convolutional neural networks on graphs with fast localized spectral filtering.

Fey, M. and Lenssen, J. E. (2019). Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*.

Hamilton, W. L., Ying, R., and Leskovec, J. (2017). Inductive representation learning on large graphs.

Hoehndorf, R., Schofield, P. N., and Gkoutos, G. V. (2011). PhenomeNET: a whole-phenome approach to disease gene discovery. *Nucleic Acids Research*, **39**(18), e119–e119.

Honda, S., Shi, S., and Ueda, H. R. (2019). Smiles transformer: Pre-trained molecular fingerprint for low data drug discovery.

Huang, K., Xiao, C., Glass, L. M., and Sun, J. (2020). MolTrans: Molecular interaction transformer for drug–target interaction prediction. *Bioinformatics*.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization.

Kipf, T. N. and Welling, M. (2016). Semi-supervised classification with graph convolutional networks.

Klicpera, J., Bojchevski, A., and Günnemann, S. (2018). Predict then propagate: Graph neural networks meet personalized pagerank.

Köhler, S., Carmody, L., Vasilevsky, N., Jacobsen, J. O. B., Danis, D., Gourdine, J.-P., Gargano, M., Harris, N. L., Matentzoglu, N., McMurry, J. A., Osumi-Sutherland, D., Cipriani, V., Balhoff, J. P., Conlin, T., Blau, H., Baynam, G., Palmer, R., Gratian, D., Dawkins, H., Segal, M., Jansen, A. C., Muaz, A., Chang, W. H., Bergerson, J., Laulederkind, S. J. F., Yüksel, Z., Beltran, S., Freeman, A. F., Sergouniotis, P. I., Durkin, D., Storm, A. L., Hanauer, M., Brudno, M., Bello, S. M., Sincan, M., Rageth, K., Wheeler, M. T., Oegema, R., Lourghi, H., Rocca, M. G. D., Thompson, R., Castellanos, F., Priest, J., Cunningham-Rundles, C., Hegde, A., Lovering, R. C., Hajek, C., Olry, A., Notarangelo, L., Similuk, M., Zhang, X. A., Gómez-Andrés, D., Lochmüller, H., Dollfus, H., Rosenzweig, S., Marwaha, S., Rath, A., Sullivan, K., Smith, C., Milner, J. D., Leroux, D., Boerkoel, C. F., Klion, A., Carter, M. C., Groza, T., Smedley, D., Haendel, M. A., Mungall, C., and Robinson, P. N. (2018). Expansion of the human phenotype ontology (HPO) knowledge base and resources. *Nucleic Acids Research*, **47**(D1), D1018–D1027.

Kuhn, M., Letunic, I., Jensen, L. J., and Bork, P. (2015). The SIDER database of drugs and side effects. *Nucleic Acids Research*, **44**(D1), D1075–D1079.

Kulmanov, M. and Hoehndorf, R. (2019). DeepGOPlus: improved protein function prediction from sequence. *Bioinformatics*.

Lee, I., Keum, J., and Nam, H. (2019). DeepConv-DTI: Prediction of drug-target interactions via deep learning with convolution on protein sequences. *PLOS Computational Biology*, **15**(6), e1007129.

Li, G., Müller, M., Thabet, A., and Ghanem, B. (2019). Deepgcns: Can gcns go as deep as cnns?

Li, G., Xiong, C., Thabet, A., and Ghanem, B. (2020a). Deepergcn: All you need to train deeper gcns.

Li, G., Xiong, C., Thabet, A., and Ghanem, B. (2020b). Deepergcn: All you need to train deeper gcns.

Luo, Y., Zhao, X., Zhou, J., Yang, J., Zhang, Y., Kuang, W., Peng, J., Chen, L., and Zeng, J. (2017). A network integration approach for drug-target interaction prediction and computational drug repositioning from heterogeneous information.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space.

Mozzicato, P. (2009). MedDRA. *Pharmaceutical Medicine*, **23**(2), 65–75.

Overington, J. P., Al-Lazikani, B., and Hopkins, A. L. (2006). How many drug targets are there? *Nature Reviews Drug Discovery*, **5**(12), 993–996.

Öztürk, H., Özgür, A., and Ozkirimli, E. (2018). DeepDTA: deep drug–target binding affinity prediction. *Bioinformatics*, **34**(17), i821–i829.

Pahikkala, T., Airola, A., Pietila, S., Shakyawar, S., Szwajda, A., Tang, J., and Aittokallio, T. (2014). Toward more realistic drug-target interaction predictions. *Briefings in Bioinformatics*, **16**(2), 325–337.

Smith, C. L. and Eppig, J. T. (2009). The mammalian phenotype ontology: enabling robust annotation and comparative analysis. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, **1**(3), 390–399.

Szklarczyk, D., Franceschini, A., Wyder, S., Forslund, K., Heller, D., Huerta-Cepas, J., Simonovic, M., Roth, A., Santos, A., Tsafou, K. P., Kuhn, M., Peer, Jensen, L. J., and von Mering, C. (2014). STRING v10: protein–protein interaction networks, integrated over the tree of life. *Nucleic Acids Research*, **43**(D1), D447–D452.

Szklarczyk, D., Santos, A., von Mering, C., Jensen, L. J., Bork, P., and Kuhn, M. (2015). STITCH 5: augmenting protein–chemical interaction networks with tissue and affinity data. *Nucleic Acids Research*, **44**(D1), D380–D384.

Thafar, M. A., Olayan, R. S., Ashoor, H., Albaradei, S., Bajic, V. B., Gao, X., Gojobori, T., and Essack, M. (2020). DTiGEMS: drug-target interaction prediction using graph embedding, graph mining, and similarity-based techniques. *Journal of Cheminformatics*, **12**(1).

Vazquez, A., Flammini, A., Maritan, A., and Vespignani, A. (2003). Global protein function prediction from protein-protein interaction networks. *Nature Biotechnology*, **21**(6), 697–700.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. (2017). Graph attention networks.

Wan, F., Hong, L., Xiao, A., Jiang, T., and Zeng, J. (2019). NeoDTI: neural integration of neighbor information from a heterogeneous network for discovering new drug–target interactions. *Bioinformatics*, **35**(1), 104–111.

Wang, C. and Kurgan, L. (2018). Review and comparative assessment of similarity-based methods for prediction of drug–protein interactions in the druggable human proteome. *Briefings in Bioinformatics*, **20**(6), 2066–2087.

Wen, M., Zhang, Z., Niu, S., Sha, H., Yang, R., Yun, Y., and Lu, H. (2017). Deep-learning-based drug–target interaction prediction. *Journal of Proteome Research*, **16**(4), 1401–1409.

Wishart, D. S., Knox, C., Guo, A. C., Cheng, D., Shrivastava, S., Tzur, D., Gautam, B., and Hassanali, M. (2007). DrugBank: a knowledgebase for drugs, drug actions and drug targets. *Nucleic Acids Research*, **36**(suppl_1), D901–D906.

Wishart, D. S., Feunang, Y. D., Guo, A. C., Lo, E. J., Marcu, A., Grant, J. R., Sajed, T., Johnson, D., Li, C., Sayeeda, Z., Assempour, N., Iynkkaran, I., Liu, Y., Maciejewski, A., Gale, N., Wilson, A., Chin, L., Cummings, R., Le, D., Pon, A., Knox, C., and Wilson, M. (2017). DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic Acids Research*, **46**(D1), D1074–D1082.

Zitnik, M. and Leskovec, J. (2017). Predicting multicellular function through multi-layer tissue networks.

Zitnik, M., Sosic, R., and Leskovec, J. (2018). Biosnap datasets: Stanford biomedical network dataset collection. *Note: http://snap. stanford. edu/biodata Cited by*, **5**(1).

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture