Computers in Biology and Medicine 150 (2022) 105992

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

A computational approach to drug repurposing using graph neural networks

Siddhant Doshi *, Sundeep Prabhakar Chepuri

Indian Institute of Science, Bangalore, 560012, India

ARTICLE INFO

Keywords:
Computational pharmacology
Drug repurposing
Drug repositioning
Graph neural networks
Link prediction

ABSTRACT

Drug repurposing is an approach to identify new medical indications of approved drugs. This work presents a graph neural network drug repurposing model, which we refer to as GDRnet, to efficiently screen a large database of approved drugs and predict the possible treatment for novel diseases. We pose drug repurposing as a link prediction problem in a multi-layered heterogeneous network with about 1.4 million edges capturing complex interactions between nearly 42,000 nodes representing drugs, diseases, genes, and human anatomies. GDRnet has an encoder–decoder architecture, which is trained in an end-to-end manner to generate scores for drug–disease pairs under test. We demonstrate the efficacy of the proposed model on real datasets as compared to other state-of-the-art baseline methods. For a majority of the diseases, GDRnet ranks the actual treatment drug in the top 15. Furthermore, we apply GDRnet on a coronavirus disease (COVID-19) dataset and show that many drugs from the predicted list are being studied for their efficacy against the disease.

1. Introduction

Drug repurposing involves strategies to identify new medical indications of approved drugs. It includes identifying potential drugs from a large database of clinically approved drugs and monitoring their *in vivo* efficacy and potency against novel diseases. Drug repurposing is a low-risk strategy as drugs to be screened have already been approved with less unknown harmful adverse effects and requires less financial investment compared to discovering new drugs [1]. Some of the successful examples of repurposed drugs in the past are *Sildenafil*, which was initially developed as an antihypertensive drug and later proved to be effective also in treating erectile dysfunction [1] and *Rituximab* that was originally used against cancer was proved to be effective against rheumatoid arthritis [1]. Even during the coronavirus disease 2019 (COVID-19) pandemic, caused by the novel severe acute respiratory syndrome coronavirus (*SARS-CoV2*), which has affected about 450 million people with more than six million deaths worldwide as of February 2022, drug repurposing has been proved very beneficial. Approved drugs like *Remdesivir* (a drug for treating Ebola virus disease), *Ivermectin* (anthelmintic drug), *Dexamethasone* (anti-inflammatory drugs) are being studied for their efficacy against the disease [2–4].

Experimental and computational approaches are usually considered for identifying the right candidate drugs, which is the most critical step in drug repurposing. To identify the candidate drugs experimentally, a variety of chromatographic and spectroscopic techniques are available for target-based drug discovery. Phenotype screening is used as an alternative to target-based drug discovery when the identity of the specific drug target and its role in the disease is not known [1]. Recently, computational approaches for identifying the candidates for drug repurposing are gaining popularity due to the availability of large biological data. Efficient ways to handle big data have opened up many opportunities in the field of pharmacology. For instance, [5] elaborates several data-driven computational tools using machine learning (ML) and deep learning (DL) techniques to integrate large volumes of heterogeneous data and solve problems in pharmacology such as drug-target interaction prediction and drug–drug interaction prediction [6], to list a few. Drug repurposing has been studied using computational methods such as signature matching methods, molecular docking, matrix factorization-based, and network-based approaches [7–13]. However, signature matching approaches and molecular docking approaches rely highly on knowing profiles and exact structures of the target genes, that may not be always available. The matrix factorization-based models find new drug–disease interactions by quantifying the similarity between drugs and disease causative viruses using their molecular sequences. However, these approaches are restricted to pairwise similarities and fail to capture the interactions at a global level [13]. The network proximity-based methods predict drugs for a disease by calculating the network proximity scores between the target genes of the drug and the target genes of the disease [9,10], but these methods cannot easily account for the additional information in the network, such as similarities between drugs or diseases. Recently, representation learning techniques (i.e., machine learning and deep learning) have been gaining attention due to their accelerated and improved

* Corresponding author.
E-mail addresses: siddhant.doshi@outlook.com (S. Doshi), spchepuri@iisc.ac.in (S.P. Chepuri).

https://doi.org/10.1016/j.compbiomed.2022.105992
Received 16 April 2022; Received in revised form 5 August 2022; Accepted 14 August 2022
Available online 31 August 2022

benefits for drug repurposing over the traditional non-deep learning methods [14,15]. Existing deep learning techniques for drug repurposing can be categorized into sequence-based methods and graph-based methods [15]. The sequence-based methods use the molecular structural sequences of drugs and the virus genome sequence of diseases to encode their respective entity-specific information [16]. However, these methods are highly dependent on the availability of the sequence information for each entity. Also, these approaches focus on the consecutive one- or two-dimensional correlation in a sequence, but do not capture the interactions at a global level between different biological entities. On the other hand, the graph-based approaches capture the structural connectivity information between different biological entities and provide more flexible framework for modeling complex biological interactions between the underlying entities [11,12,17].

A natural and efficient way to capture complex interactions between different biological entities like drugs, genes, diseases, etc., is to construct a graph with nodes representing entities and edges representing interactions between these entities, e.g., interactions between drugs and genes or between drugs and diseases. The graph-based methods such as the deepwalk-based, or graph neural networks, that are capable of processing such graph structured biological data have been proposed for drug repurposing [11,12,17]. The deepwalk-based architecture [17] independently generates the structural information (using the deepwalk algorithm) and the self entity information due to which the entity and the relational correspondence is not well captured. *Graph neural networks* (GNNs) capture structural information in data by accounting for interactions between various underlying entities while processing data associated with them, thus producing meaningful low-dimensional embeddings for the entities that are useful for downstream machine learning tasks. However, the existing GNN-based models have a considerable computational overhead when processing huge biological networks having interactions of high density. In this work, we address this problem and focus on drug repurposing using computationally-efficient GNNs. We provide a comparative analysis of several graph-based architectures for drug repurposing and showcase the benefits of having a dedicated model through our experiments on real datasets.

1.1. Main results and contributions

We construct a four-layered heterogeneous graph explaining interactions between the four entities, namely, drugs, genes, diseases, and anatomies in each layer. We propose a new dedicated GNN model for drug repurposing, called GDRnet, which has an encoder–decoder architecture. We formulate drug repurposing as a link prediction problem and train GDRnet to predict unknown links between the drug and disease entities, where a link between a drug–disease entity suggests that the drug treats the disease. Specifically, the encoder is based on the scalable inceptive graph neural network (SIGN) architecture [18] for generating the node embeddings of the entities. We propose a learnable quadratic norm scoring function as a decoder to rank the predicted drugs. The proposed norm scorer is particularly designed and tuned for the drug repurposing task that learns correlations between the drug and disease pairs. The main contributions and results are summarized as follows.

- We formulate drug repurposing as a link prediction problem and propose a new dedicated GNN-based drug repurposing model. The trainable encoder of GDRnet precomputes the neighborhood features beforehand, thus, is computationally efficient with reduced training and inference time. The trainable decoder scores a drug–disease pair based on the low-dimensional embeddings obtained from the encoder. The encoder and decoder are trained in an end-to-end manner.
- We validate GDRnet in terms of its link prediction accuracy and how well it ranks the known treatment drug. For a majority of diseases with known treatment in the test set, which were not used while training, GDRnet ranks the approved treatment drugs in the top 15. This suggests the efficacy of the proposed drug repurposing model.
- We perform an ablation study to show the importance of genes and anatomy entities, which model the indirect interactions between the drug and the disease entities.
- We provide a detailed computational runtime analysis of the proposed GDRnet architecture against the existing GNN models. We demonstrate the advantage of using SIGN as an encoder in GDRnet through the performance gain achieved in terms of its training and inference time.
- We apply GDRnet for COVID-19 drug repurposing by including the COVID-19 interactome information from [19] in the dataset. Many of the drugs predicted by GDRnet for COVID-19 are being studied for their efficacy against the disease.

The software to reproduce the results are available in the github repository: https://github.com/siddhant-doshi/GDRnet

2. Multilayered drug repurposing graph

In this section, we model the biological data as a multilayer graph to capture the complex interactions between different biological entities. We consider four entities that are relevant to the drug repurposing task. The four entities are drugs (e.g., *Dexamethasone*, *Sirolimus*), diseases (e.g., *Scabies*, *Asthma*), anatomies (e.g., *Bronchus*, *Trachea*), and genes[1] (e.g., *DUSP11*, *PPP2R5E*). We form a four-layered heterogeneous graph with these entities as layers; see the illustration in Fig. 1a.

In the multilayer graph, i.e., the interactome there are inter-layered connections between the four layers and intra-layered connections within each layer. The inter-layered connections are of different types. The drug–disease links indicate treatment or palliation, i.e., a drug treats or has a relieving effect on a disease. For example, interaction between *Ivermectin-Scabies* (as seen in Fig. 1b) and *Simvastatin-Hyperlipidemia* (as seen in Fig. 1d) are of type treatment, whereas *Atropine-Parkinson's disease* is of type palliation. The drug–gene and disease–gene links are the direct gene targets of the compound and the disease, respectively. *NR3C2*, *RHOA*, *DNMT1* are some of the target genes of the drug *Dexamethasone* (see Fig. 1b) and *PPP1R3D*, *CAV3* are target genes of the disease *Malaria*. There are also indirect links between target genes of a drug and a disease, referred to as the shared target genes (see Fig. 1b). For example, genes like *ATF3*, *UPP1*, *CTSD*, are the shared target genes of drug *Ivermectin* and disease *Malaria*. The disease–anatomy and gene–anatomy connections indicate how the diseases affect the anatomies and interactions between the genes and anatomies. For example, *GNAI2* and *HMGCR* belong to the *cardiac ventricle* anatomy (see Fig. 1d); disease *Schizophrenia* affects multiple anatomies like the *central nervous system (CNS)* and *optic tract*.

The intra-layered drug–drug and disease–disease connections show the similarity between a pair of drugs and diseases, respectively. The gene–gene links describe the interaction between genes (e.g., epistasis, complementation) and form the whole gene interactome network. The anatomy information helps by focusing on the local interactions of genes related to the same anatomy as the genes targeted by the new disease. Some examples of the intra-layered connections are *Simvastatin-Lovastatin* and *POLA2-RAE1* as seen in Fig. 1d. This comprehensive network serves as a backbone for our model, which predicts the unknown inter-layered links between drugs and novel diseases by leveraging the multi-layered graph-structured data.

[1] All the genes are represented using the symbols according to the HUGO gene nomenclature committee (HGNC) [20].

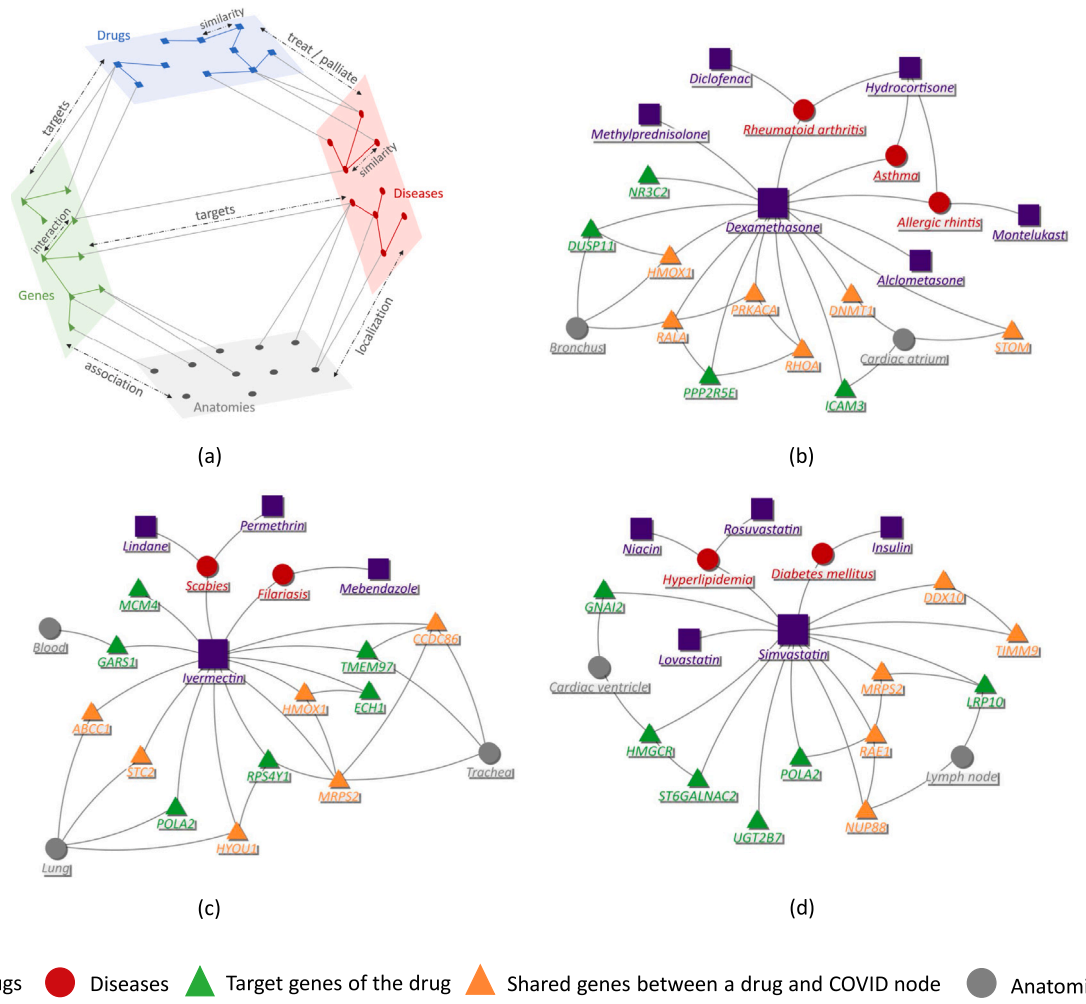

Fig. 1. Drug repurposing network. (a) Illustration of the four-layered heterogeneous graph with the inter-layer and the intra-layer connections. (b), (c) and (d) Subgraphs centered around the drugs *Dexamethasone*, *Ivermectin* and *Simvastatin*, respectively, illustrate shared target genes between these drugs and COVID-19 disease nodes (see description later on in Section 4.8).

3. Methods and models

Graph neural networks (GNNs) have become very popular for processing and analyzing such graph-structured data in the last few years. Compared to deep learning models such as convolutional neural networks (CNNs), GNNs offer extraordinary performance improvements while dealing with graph-structured data commonly encountered in social networks, biological networks, brain networks, and molecular networks, to name a few. GNN models learn low-dimensional graph representations or node embeddings that capture the nodal connectivity information useful for solving graph analysis tasks like node prediction, graph classification, and link prediction. In this section, we describe the proposed GDRnet architecture for drug repurposing, which is formulated as a link prediction problem.

3.1. Notation

Consider an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with a set of vertices $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ and edges $e_{ij} \in \mathcal{E}$ denoting a connection between nodes v_i and v_j. We represent a graph $\mathcal{G}$ using the adjacency matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$, where the (i,j)th entry of $\mathbf{A}$, denoted by a_{ij}, is 1 if there exists an edge between nodes v_i and v_j, and *zero* otherwise. To account for the non-uniformity in the degrees of the nodes, we use the normalized adjacency matrix denoted by $\tilde{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$, where $\mathbf{D} \in \mathbb{R}^{N\times N}$ is the diagonal degree matrix. Each node in the graph has attributes (referred to as input features). Let us denote the input feature vector of node v_i by $\mathbf{x}_i^{(0)} \in \mathbb{R}^d$, which contains attributes of that node.

3.2. Graph neural networks

In most of the existing GNN architectures, the embedding of a node is updated during training by sequentially aggregating information from its 1-hop neighbor nodes, thereby accounting for local interactions in the network. This is also referred to as a GNN layer. Several such GNN layers are cascaded to capture interactions beyond the 1-hop neighborhood. Specifically, by cascading K such layers, node features from its K-hop neighborhood are captured. For example, in Fig. 1c, the drug *Ivermectin* is a 2-hop neighbor of the anatomy *Lung* and is connected via *STC2*. Mathematically, the node feature vector updates can be represented by the recursion

$$\mathbf{x}_i^{(k+1)} = g_k\left(\mathbf{x}_i^{(k)}, f_k\left(\left\{\mathbf{x}_j^{(k)},\ \forall j \in \mathcal{N}_{v_i}^{(1)}\right\}\right)\right), \tag{1}$$

where $\mathbf{x}_i^{(k)} \in \mathbb{R}^{d_k}$ is the embedding for node v_i at the kth layer and $\mathcal{N}_{v_i}^{(j)}$ represents a set of j-hop neighbor nodes of node v_i. Local aggregation function $f_k(\cdot)$ combines the neighbor node features (during the training) and $g_k(\cdot)$ transforms it to obtain the updated feature vector. Different choices of the aggregation function $f_k(\cdot)$ and the transformation function $g_k(\cdot)$ lead to different GNN variants like the graph convolutional networks (GCN) [21], GraphSAGE [22], and graph attention networks (GAT) [23], to name a few. However, these GNN

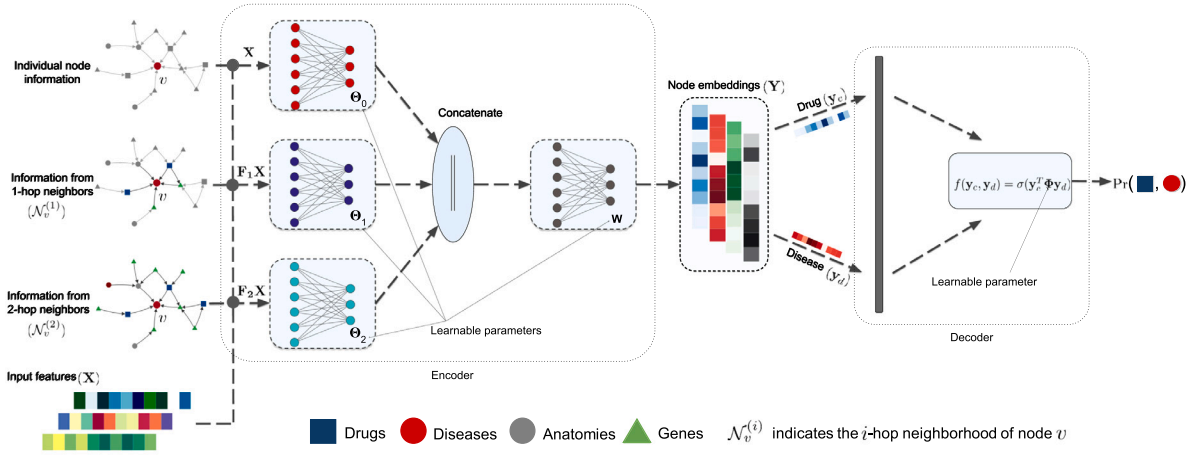

Fig. 2. The GDRnet architecture.

models do not scale well on large and dense graphs as their computational cost depends on the number of nodes and edges in the graph. To reduce the runtime computations, a scalable GNN architecture called SIGN [18] has been proposed, where the neighborhood aggregations at various depths (till K-hop) are precomputed (before training), and the node embeddings are generated non-iteratively, unlike the GNN models in Eq. (1). As the node features updates are performed beforehand outside the training procedure, these GNN variants easily scale on large graphs, such as the multi-layered drug repurposing graph, as they are independent of the number of edges in the graph. The proposed GDRnet architecture has an encoder–decoder architecture, wherein the encoder is based on the SIGN architecture due to its computational advantages. While SIGN has been used for node classification [18], we utilize it here for link prediction, i.e., to predict links between drugs and diseases. Next, we describe the proposed GDRnet architecture.

3.3. The GDRnet architecture

The proposed GNN architecture for drug repurposing has two main components, namely, the encoder and decoder. The encoder generates the node embeddings of all the nodes in the four-layer graph. The decoder scores a drug–disease pair based on the embeddings. The encoder and decoder networks are trained in an end-to-end manner. Next, we describe these two components of the GDRnet architecture, which is illustrated in Fig. 2.

3.3.1. Encoder

The GDRnet encoder produces low-dimensional node embeddings based on the input features and nodal connectivity information. Recall that the matrix $\tilde{\mathbf{A}}$ is the normalized adjacency matrix of the four-layered graph $\mathcal{G}$. We use graph operators represented using matrices $\mathbf{F}_r = \tilde{\mathbf{A}}^r$, $r = 1, 2, \ldots$, to aggregate information in the graph. Here, $\tilde{\mathbf{A}}^r$ denotes the rth matrix power. By choosing $\mathbf{F}_r = \tilde{\mathbf{A}}^r$, we aggregate information from the r-hop neighborhood. We assume that each node has its own d-dimensional feature, which we collect in the matrix $\mathbf{X} \in \mathbb{R}^{N\times d}$ to obtain the input feature matrix associated with the nodes of $\mathcal{G}$. We can then represent the encoder as

$$\mathbf{Z} = \sigma_1\left\{\left[\mathbf{X}\boldsymbol{\Theta}_0 \,\|\, \mathbf{F}_1\mathbf{X}\boldsymbol{\Theta}_1 \| \ldots \| \,\mathbf{F}_r\mathbf{X}\boldsymbol{\Theta}_r\right]\right\} \quad \text{and} \quad \mathbf{Y} = \sigma_2\left\{\mathbf{Z}\mathbf{W}\right\}, \tag{2}$$

where $\mathbf{Y}$ is the final node embedding matrix for the nodes in the graph $\mathcal{G}$ and $\{\boldsymbol{\Theta}_0, \ldots, \boldsymbol{\Theta}_r, \mathbf{W}\}$ are the learnable parameters. Here, $\|$ represents concatenation, $\sigma_1\{\cdot\}$ and $\sigma_2\{\cdot\}$ are the nonlinear tanh and leaky rectified linear unit (leaky ReLU) activation functions, respectively. The matrix $\mathbf{F}_r\mathbf{X} = \tilde{\mathbf{A}}^r\mathbf{X}$ aggregates node features from r-hop neighbors, which can be related to the neighborhood aggregation performed at the rth layer of GNN models that perform sequential neighborhood aggregation as in Eq. (1). Fig. 2 shows the encoder architecture. The main advantage of using SIGN over other models (e.g., GCN, GAT, GraphSAGE) is that the matrix product $\mathbf{F}_r\mathbf{X}$ is independent of the learnable parameters $\boldsymbol{\Theta}_r$. Thus, this matrix product can be precomputed before training the neural network model. Doing so reduces the computational complexity while incorporating information from the graph structure.

In our experiments, we choose $r = 2$, i.e., the low-dimensional node embeddings have information from 2-hop neighbors. Choosing $r \geq 3$ is found to be not useful for drug repurposing, as we aim to capture the local information of the drug targets such that a drug node embedding should retain information about its target genes and the shared genes in its vicinity. For example, the 1-hop neighbors of *Dexamethasone* as shown in Fig. 1b, are the diseases it treats (e.g., *Asthma*), and the drugs similar to *Dexamethasone* (e.g., *Methylprednisolone*) and its target genes (e.g., *DUSP11*, *RHOA*). The 2-hop neighbors are the anatomies of the target genes (e.g., *Bronchus*), and the drugs that have similar effects on the diseases (e.g., *Hydrocortisone* and *Dexamethasone* have similar effects on *Asthma*). While updating the node for the embedding related to *Dexamethasone*, it is important to retain this local information for the drug repurposing task.

3.3.2. Decoder

For drug repurposing, we propose a score function based on a general dot-product that takes as input the updated embeddings of drugs and diseases and outputs a score based on which we decide if a certain drug treats the disease. Fig. 2 illustrates the proposed learnable decoder. The columns of the embedding matrix $\mathbf{Y}$ contain the embeddings of all the nodes in the four-layer graph, including the embeddings of the disease and drug nodes. Let us denote the embeddings of the ith drug as $\mathbf{y}_{c_i} \in \mathbb{R}^l$ and the embeddings of the jth disease as $\mathbf{y}_{d_j} \in \mathbb{R}^l$. The proposed scoring function $\texttt{score}(\cdot)$ to infer whether drug c_i is a promising treatment for disease d_j is defined as

$$s_{ij} = \texttt{score}\left(\mathbf{y}_{c_i}, \mathbf{y}_{d_j}\right) = \sigma\left\{\mathbf{y}_{c_i}^T\boldsymbol{\Phi}\mathbf{y}_{d_j}\right\}, \tag{3}$$

where $\sigma\{\cdot\}$ is the nonlinear sigmoid activation function and $\boldsymbol{\Phi} \in \mathbb{R}^{l\times l}$ is a learnable co-efficient matrix. We interpret s_{ij} as the probability that a link exists between drug c_i and disease d_j. The term $\mathbf{y}_{c_i}^T\boldsymbol{\Phi}\mathbf{y}_{d_j}$ can be interpreted as a measure of correlation (induced by $\boldsymbol{\Phi}$) between the disease and drug node embeddings.

3.3.3. Training loss

The model is trained in a mini-batch setting in an end-to-end fashion using stochastic gradient descent to minimize the weighted

Table 1
Multi-layered graph data. The value in each cell represents the number of links between the respective layers. NC represents no connection.

Drugs	6486			
Diseases	6113	543		
Genes	76 250	123 609	474 526	
Anatomies	NC	3602	726 495	NC
	Drugs	Diseases	Genes	Anatomies

cross-entropy loss, where the loss function for the sample corresponding to the drug–disease pair (i, j) is given by

$$\ell(s_{ij}, z_{ij}) = w\, z_{ij} \left(\log \left(\frac{1}{\sigma(s_{ij})} \right) \right) + \left(1 - z_{ij}\right) \log \left(\frac{1}{1 - \sigma(s_{ij})} \right), \tag{4}$$

where z_{ij} is the known training label associated with the score s_{ij} for the drug–disease pair $(c_i,\ d_j)$, $z_{ij} = 1$ indicates that drug i treats or palliates disease j, and $z_{ij} = 0$ otherwise. Here, w is the weight on the positive samples that we choose to account for the huge class imbalance in the dataset. During training, we include no-drug–disease links, which give us the negative control for learning. For example, there is no link between the drug–disease pair *Simvastatin-Scabies*, i.e., *Simvastatin* is not known to treat or suppress the effects of *Scabies*. The number of no-drug–disease links is almost thirty times the number of positive samples. To handle this class disparity, we explicitly use a weight $w > 0$ on the positive samples.

4. Model evaluation and experiments

In this section, we evaluate GDRnet and discuss the choice of various hyper-parameters. The model is evaluated based on two performance measures. Firstly, we report the ability to classify the links correctly, i.e., to predict the known treatments correctly for diseases in the test set. Next, using the list of predicted drugs for the diseases in the test set, we report the model's ability to rank the actual treatment drug as high as possible (the ranking is obtained by ordering the scores in Eq. (3)). Finally, we also report prediction results for coronavirus related diseases.

4.1. Dataset

We use information from the drug repurposing knowledge graph (DRKG) [24] to form the multi-layered drug repurposing graph. DRKG includes information about six drug databases, namely, Drugbank [25], Hetionet [26], GNBR [27], STRING [28], IntAct [29], and DGIdb [30]. We construct a four-layered graph comprising the drug layer, disease layer, gene layer, and anatomy layer. We extract the details about these entities specifically from the Drugbank, Hetionet, and GNBR databases. We leverage their generic set of low-dimensional embeddings that represent the graph nodes and edges in the Euclidean space for training. The four-layered graph is composed of 8070 drugs, 4166 diseases, 29 848 genes, 400 anatomies, and a total of 1,417,624 links, which include all the inter-layer and intra-layer connections (refer Section 2 for the description of the multi-layered graph). Details about the inter-layered and intra-layered links are given in Table 1.

4.2. Experimental setup and model parameters

The drug repurposing problem is formulated as a link prediction. It can be viewed as a binary classification problem, wherein a positive class represents the existence of a link between a drug and disease, and otherwise represents a negative class. We have 6113 positive samples (drug–disease links) in our dataset. To account for the negative class samples, we randomly choose 200,000 no-drug–disease links (i.e., those pairs with no link between these drugs and diseases). These links are then divided into the training and testing set with a 90%−10% split. We train the network using mini-batch stochastic gradient descent by grouping the training set in batches of size 512 and train them for nearly 20 epochs. Due to the significant class imbalance, we oversample the drug–disease links while creating batches, thus maintaining the class ratio (ratio of the number of negative samples to the number of positive samples) of 1.5 in each batch. The additional hyperparameters are set as follows. The intermediate embedding dimensions are fixed to 250, the batch size and the learning rate (set to 10^{-4}) are chosen by performing a grid search over the hyperparameter space. Also, we use the leaky rectified linear unit (Leaky-ReLU) as the intermediate activation function. We use the Adam optimizer to perform the back propagation and update the model parameters. The weight w on the positive samples (cf. Eq. (4)) is also chosen to be the class imbalance ratio of each batch, i.e., we fix w to be 1.5.

4.3. Baselines

We perform experiments on the state-of-the-art network-based drug repurposing methods, the network-proximity based [9], which is based on the Z-scores computed using the permutation test, the HINGRL [17] method based on the autoencoder and deepwalk algorithm, and the Bipartite-GCN method [31], which uses an attention-based GNN layer. In addition, we also provide a comparison with three commonly used GNN encoder architectures, namely, GCN [21], GraphSAGE [22], and GAT [23] for the drug repurposing task, which we treat as a link prediction problem, and compare the classification performance with the GDRnet architecture. Specifically, the encoder in GDRnet is replaced with GCN, GraphSAGE, and GAT to evaluate the model performance. Two blocks of these sequential models are cascaded to maintain consistency with $r = 2$ of the GDRnet architecture. We evaluate these models on the test set, which contains known treatments for diseases that are not shown to the model while training. To remain consistent, we use the same initial embeddings for all the experiments.

4.4. Classification performance

We measure the classification abilities of a model through the receiver operating characteristic (ROC) curve of the true positive rate (TPR) versus the false positive rates (FPR) and the precision–recall (PR) curve of the precision versus the recall. The area under the PR curves (AUPRC) along with the area under the receiver operating characteristics (AUROC), would give a comprehensive view of the performance statistics of the encoders. Fig. 3a shows the ROC curves of different GNN models. We can see that all the models have very similar AUROC values. Also, all the AUPRC values, as shown in Fig. 3b are in a similar range. As compared to the baseline precision of 0.03, which is calculated as the ratio of the minority class in the data, we see a significant gain in the AUPRC values. Fig. 4a provides an illustration of two-dimensional embeddings (from GDRnet), using the t-distributed stochastic neighbor embedding (t-SNE), where we observe that diseases that target certain anatomy or a drug that target certain gene have nearby representations in the embedding space demonstrating the expressive power of GDRnet.

4.5. Ranking performance

We evaluate GDRnet in terms of ranks of the actual treatment drug in the predicted list for a disease from the testing set, where the rank is computed by rank ordering the scores. Fig. 5 represents the histograms of the ranks of the drug–disease pairs from the testing set for GraphSAGE, GCN, GAT, HINGRL, and Bipartite-GCN compared with GDRnet. To get the histograms, we compute the ranks of the actual treatment drugs for the diseases from the test set and plot the frequencies of those ranks on the vertical axis corresponding to the ranks on the horizontal axis. We see that GDRnet has a higher density of ranks in the top

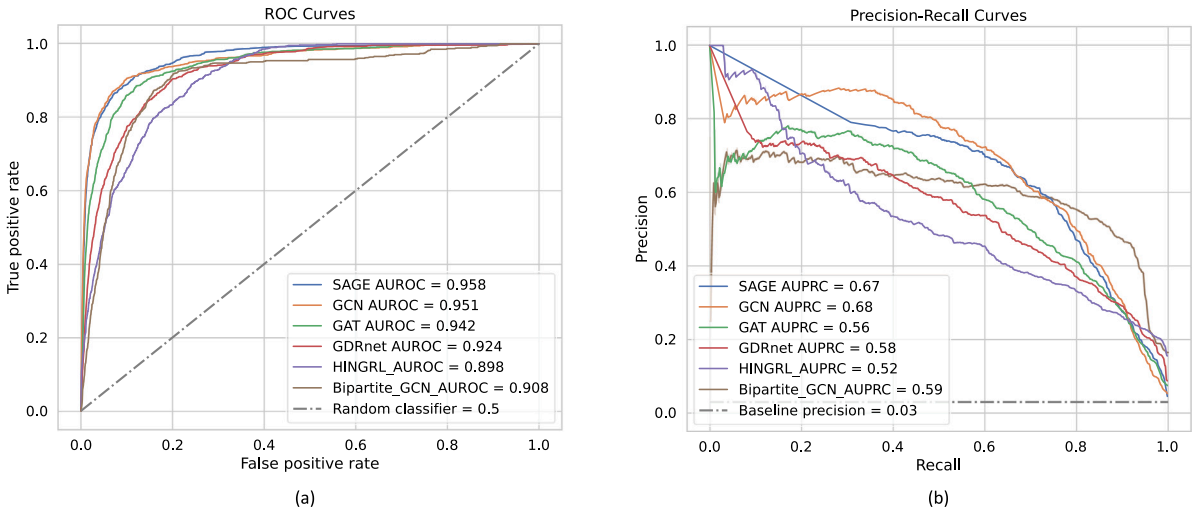

Fig. 3. Classification performance of GDRnet. (a) and (b) represent the receiver operating curves (ROC) and the precision–recall (PR) curves, respectively, depicting the classification performance of different drug-repurposing models.

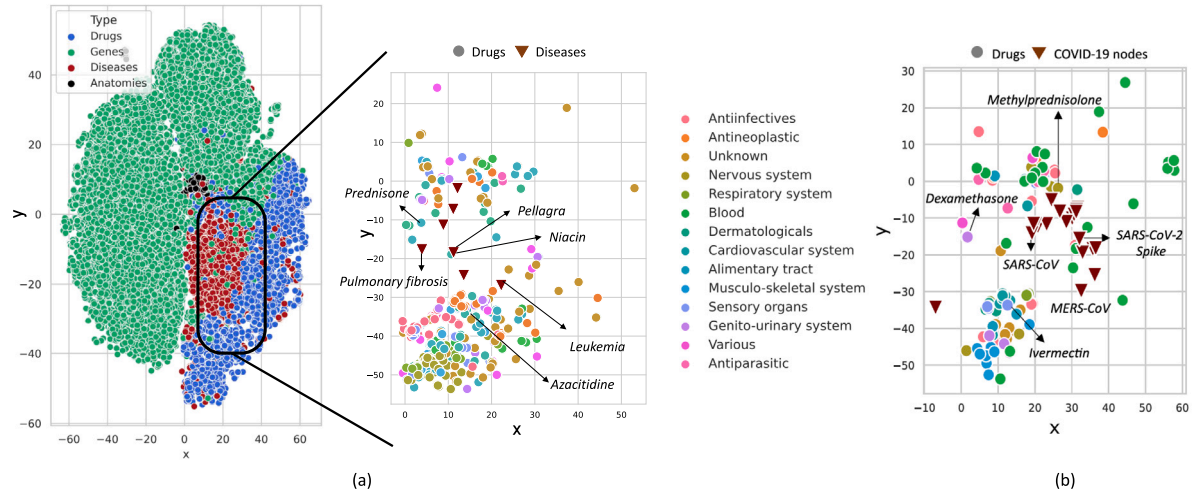

Fig. 4. Embedding visualization. (a) Two-dimensional t-SNE visualization of the high-dimensional embeddings generated by GDRnet for the nodes in the four-layered heterogeneous graph. The left embedding plot shows the representation of all the nodes (around 42 000 nodes), which are colored according to their layer. The right plot focuses on the drugs and diseases used for testing. (b) Embeddings of the COVID-19 disease nodes (27 SARS-CoV-2 proteins and 6 coronavirus related diseases) and the predicted drugs by GDRnet. The drugs in the both the plots (a) and (b) are colored according to their first-level anatomical therapeutic chemical (ATC) categorization.

15 as compared to other models. This clearly illustrates that GDRnet outperforms the other graph-based methods in terms of its ranking abilities. In addition, we compute the network proximity scores [9] and rank order the drugs based on network proximity scores to compare with the GNN-based encoder models. These network proximity scores are a measure of the shortest distance between drugs and diseases through their target genes. They are computed as

$$P_{ij} = \frac{1}{|\mathcal{C}| + |\mathcal{T}|}\left(\sum_{p\in\mathcal{C}} \min_{q\in\mathcal{T}} d(p,q) + \sum_{q\in\mathcal{T}} \min_{p\in\mathcal{C}} d(p,q)\right), \tag{5}$$

where P_{ij} is a proximity score of drug c_i and disease d_j. Here, $\mathcal{C}$ is the set of target genes of c_i, $\mathcal{T}$ is the set of target genes of d_j, and $d(p,q)$ is the shortest distance between a gene $p \in \mathcal{C}$ and a gene $q \in \mathcal{T}$ in the gene interactome. We convert these into Z-scores using the permutation test $Z_{ij} = (P_{ij} - \mu)/\omega$, where μ is the mean proximity score of the pair (c_i, d_j) computed by randomly selecting subsets of genes with the same degree distribution as that of $\mathcal{C}$ and $\mathcal{T}$ from the gene interactome, and ω is the standard deviation of the scores generated in the permutation test of these randomly selected subsets. Table 2 provides the rankings of a few sample drug–disease pairs from the test set that were not shown during the training. We can see that the GDRnet and the other GNN variants result in better ranks on the unseen diseases than the network proximity measure, which is solely based on the gene interactome, by a huge margin. Also, determining the network proximity scores is extremely computationally expensive due to the calculation of Z-scores using the permutation test. For the same reasons we leave off the histogram analysis for the network proximity approach, which evidently through the examples in Table 2, results in poor ranking performance. The diseases on which we evaluate are not confined to a single anatomy (e.g., *rectal neoplasms* are associated to the *rectum* anatomy, whereas *pulmonary fibrosis* is a *lung* disease), nor do they indicate a similar family of drugs for their treatment (e.g., *Fluorouracil* is an antineoplastic drug, and *Prednisone* is an anti-inflammatory corticosteroid). For a majority of the diseases in the test set, GDRnet ranks the treatment drug in the top 15 (as seen in Table 2). In the case of *Leukemia*, other antineoplastic drugs like *Hydroxyurea* and *Methotrexate* are ranked high (in top 10) and its known treatment drug *Azacitidine* is ranked 17.

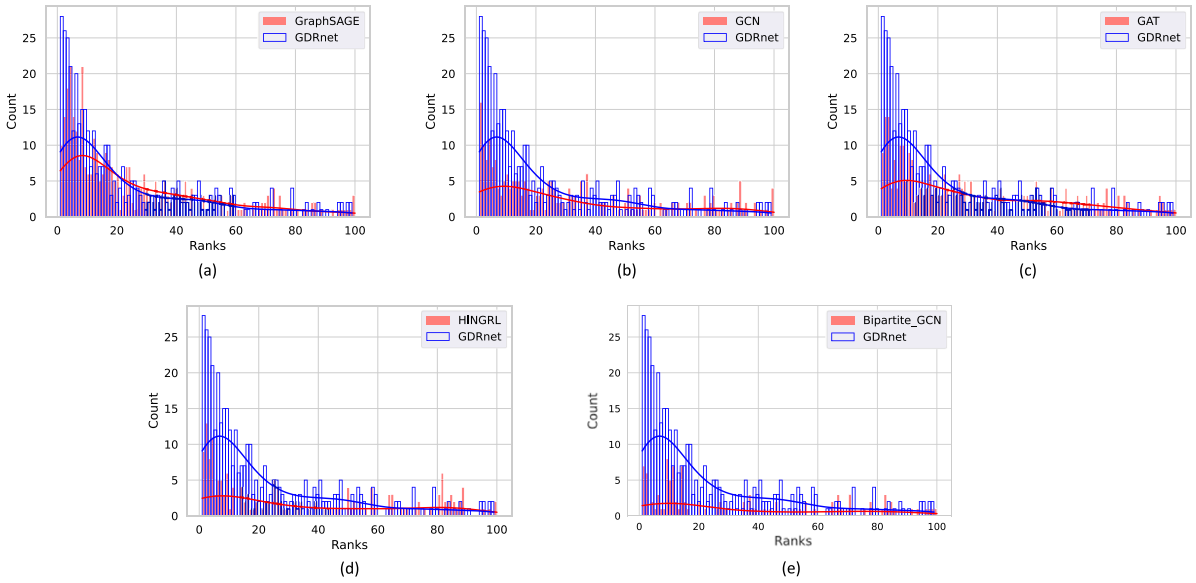

Fig. 5. Ranking histograms. The ranking performance of GDRnet compared with other GNN variants, namely, (a) GraphSAGE (b) GCN and (c) GAT (d) HINGRL (e) Bipartite-GCN.

Table 2
Ranking. A few examples of the ranks of the actual treatment drugs for the diseases from the testing set. There are no associated genes with some of the disease in our database, which makes it impossible to rank them using the network proximity based method. These are indicated as "Not computable". The best ranks are highlighted in **bold**.

Disease	Treatment drug	Ranks						
		GDRnet	GraphSAGE	GCN	GAT	Network proximity	HINGRL	Bipartite-GCN
Encephalitis	*Acyclovir*	**10**	35	35	295	5462	435	27
Rectal neoplasms	*Fluorouracil*	**9**	421	16	231	2831	205	117
Pulmonary fibrosis	*Prednisone*	5	3	10	9	2072	**2**	9
Atrioventricular block	*Atropine*	**6**	79	8	14	4453	26	196
Pellagra	*Niacin*	**2**	56	497	484	Not computable	460	288
Colic	*Hyoscyamine*	**1**	**1**	501	205	Not computable	39	101
Leukemia	*Azacitidine*	**17**	120	31	332	377	527	507

Table 3
Layer ablation study. The AUROC values for a link prediction task compared across different graph layers and different GNN models. Best performances are indicated in **bold**.

Graph layers	GDRnet	GraphSAGE	GCN	GAT
Drugs, Diseases	0.61 ± 0.02	0.707 ± 0.02	0.692 ± 0.02	0.655 ± 0.01
Drugs, Diseases, Anatomies	0.652 ± 0.01	0.75 ± 0.01	0.728 ± 0.02	0.722 ± 0.01
Drugs, Diseases, Genes	0.845 ± 0.02	**0.881 ± 0.02**	0.833 ± 0.01	0.84 ± 0.01
Drugs, Diseases, Genes, Anatomies	**0.855 ± 0.02**	0.874 ± 0.01	**0.842 ± 0.02**	**0.863 ± 0.01**

4.6. Layer ablation study

To gain more insights on the importance of different entities, namely, drugs, disease, genes, and anatomies for drug repurposing, we perform an ablation study on the layers of the constructed graph. We perform link prediction using considered GNN models on the constructed graphs, starting with the only drug–disease two-layered graph, followed by the individual addition of the gene and the anatomy interactome, making it a three-layered graph, and eventually converting it to a four-layered graph by getting all the layers together. We report the corresponding AUROC values in Table 3. We use the degree information as the input features for these experiments to eliminate any biases due to the pre-trained embeddings. As seen in Table 3, the addition of the anatomy and the gene layer shows their importance by giving a significant improvement in the classification performance, demonstrating the significance of the indirect connections provided by the anatomy and the gene layers between the drugs and diseases for drug repurposing. Finally, when all the information from the four layers used together, we see a clear boost in the performance.

In summary, GNNs perform better than the prior network-based approaches in predicting the drugs for a disease. This also signifies the importance of capturing the local interactions in complex biological networks. These interactions are not sufficiently captured by the network proximity methods that restrict their focus only on the target genes of a drug and a disease. The proposed GNN-based GDRnet architecture is computationally attractive and better ranks known treatment drugs for diseases than the popular sequential GNN variants.

4.7. Computational complexity

The time complexity of GNNs that perform aggregation sequentially like GCN, GraphSAGE, and GAT, is $\mathcal{O}(LNd^2+L|\mathcal{E}|d)$ for a graph having N nodes and $|\mathcal{E}|$ edges with L sequential aggregation iterations [32]. The intermediate embedding dimensions are assumed to be d. Here, the term Nd^2 corresponds to the feature transformation, and $|\mathcal{E}|d$ is the additional computations performed to identify the neighborhood for local aggregation during the training. GDRnet benefits itself in terms of the training and inference time due to its parallel framework by precomputing this neighborhood aggregations. This results in the runtime to be independent of the number of edges in the graph, having a time complexity of $\mathcal{O}(LNd^2)$, where L is the number of parallel branches. Fig. 6 illustrates the dependence of GNNs on the number of edges.

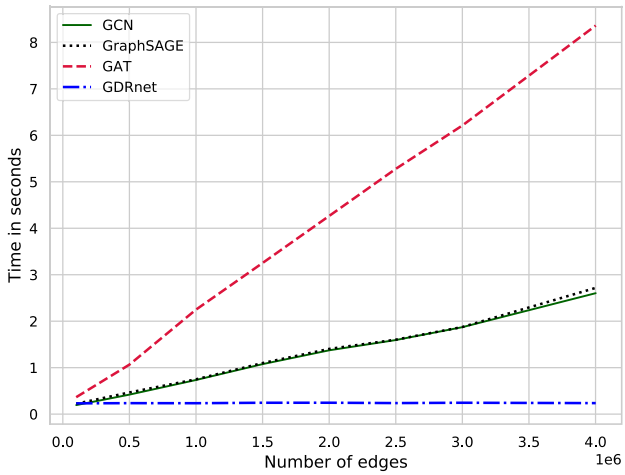

Fig. 6. Computational complexity. Time plot showing the dependence of GNN architectures on the number of edges.

Table 4
Drugs predicted by GDRnet for COVID-19.

COVID-19 node	Drugs predicted by GDRnet ranked in top 10
SARS-CoV2-E	*Ivermectin, Spironolactone, Sirolimus*
SARS-CoV2-M	*Ivermectin, Cyclosporine, Acyclovir*
SARS-CoV2-N	*Rubella virus vaccine, Sirolimus, Hydralazine*
SARS-CoV2-spike	*Crizanlizumab, Cyclosporine, Cidofovir, Nitazoxanide*
CoV-NL63	*Dexamethasone, Prednisolone, Celecoxib*

The time taken for a single epoch (forward pass) on a graph having the same number of nodes as in the constructed multilayered graph in Section 2 (approximately 42 000) are plotted on the vertical axis for varying number of edges on the horizontal axis. GCN, GraphSAGE, and GAT clearly depict their linear dependence on $|\mathcal{E}|$, whereas GDRnet verifies its independence by having a constant time, irrespective of the number of edges. The Bipartite-GCN architecture uses an attention-based graph layer similar to GAT. Thus it has the same complexity as the sequential based GNNs. It is not straightforward to compare the forward pass time complexity incurred by network proximity and HINGRL methods. HINGRL pipeline involves multiple algorithms that are trained independently, like the autoencoder, followed by deepwalk, and finally the random forests, that incur more time complexity as observed during our numerical experiments. For the network proximity method, due to the involvement of the permutation test, it is extremely computationally expensive as well.

4.8. COVID-19 drug repurposing

Next, we focus on drug repurposing for the four known human coronaviruses (HCoVs), namely, *SARS-CoV*, *MERS-CoV*, *CoV-229E* and *CoV-NL63*, and two non-human coronaviruses, namely *MHV*, and *IBV*. We consider interactions of these disease nodes with human genes. There are 129 known links between these six disease nodes and gene nodes in the dataset [24]. In addition, we consider all the 27 *SARS-CoV-2* proteins that include 4 structural proteins, namely, envelope (*SARS-CoV2-E*), membrane (*SARS-CoV2-M*), nucleocapsid (*SARS-CoV2-N*) and surface (*SARS-CoV2-spike*), 15 non-structural proteins (nsp) and 8 open reading frames (orf), and their 332 links connecting the target human genes [19]. We refer to these 33 nodes (6 disease nodes and 27 *SARS-CoV2* proteins) as the COVID-19 nodes. In other words, there are only disease–gene interactions available for these COVID-19 nodes. Some of the genes targeted by the COVID-19 nodes are shown in Fig. 1 (b, c and d), which are also the target genes for the drugs (e.g., *Dexamethasone*, *Ivermectin*, *Simvastatin*).

We individually predict the drugs for all these 33 COVID-19 nodes as each protein in *SARS-CoV-2* targets a different set of genes in humans. We select the top 10 ranked predicted drugs out of 8070 clinically approved drugs for each disease entity. Table 4 lists some of the predicted drugs by GDRnet. A complete list of the predicted drugs with their scores and ranks is available in our repository at: https://github.com/siddhant-doshi/GDRnet. Our predictions have corticosteroids like *Dexamethasone, Methylprednisolone*, antineoplastic drugs like *Sirolimus, Anakinra*, anti-parasitic drugs like *Ivermectin, Nitazoxanide*, non-steroidal anti-inflammatory drugs (NSAIDs) like *Ibuprofen, Celecoxib*, ACE inhibitors and statin drugs like *Simvastatin, Atorvastatin*, and some of the vaccines discovered previously for other diseases like the *Rubella virus vaccine*. Fig. 4b gives a two-dimensional t-SNE representation of the embeddings of a few predicted drugs and the COVID-19 disease nodes, where we can see that the representation of the predicted drugs is in the vicinity of the disease nodes in the embedding space.

5. Conclusions and future work

We proposed a GNN model for drug repurposing model, called GDRnet, to predict drugs from a large database of approved drugs for further studies. We leverage a biological network of drugs, diseases, genes, and anatomies and cast the drug repurposing task as a link prediction problem. The proposed GDRnet architecture has a computationally attractive encoder to generate low-dimensional embeddings of the entities and a decoder that scores the drug–disease pairs. Through numerical simulations on real data, we demonstrate the efficacy of the proposed approach for drug repurposing. We also apply GDRnet on COVID-19 data.

This work can be extended along several directions. Considering the availability of substantial biological data, the inclusion of information like individual side effects of drugs, may further improve the predictions. Considering the comorbidities of a patient would help us analyze the biological process and gene interactions in the body specific to an individual and accordingly prescribe the line of treatment. Also, including the edge specific information such as type of drug interactions could help us predicting a synergistic combination of drugs for a disease.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

S.P. Chepuri is supported in part by the Pratiskha Trust Young Investigator Award, Indian Institute of Science, Bangalore, and the SERB, India grant SRG/2019/000619, and S. Doshi is supported by the Robert Bosch Center for Cyber Physical Systems, Indian Institute of Science, Bangalore, Student Research Grant 2020-M-11. The authors thank the Deep Graph Learning team for making DRKG public at https://github.com/gnn4dr/DRKG.

References

[1] S. Pushpakom, F. Iorio, P.A. Eyers, K.J. Escott, S. Hopper, A. Wells, T. Doig, J. Latimer, C. McNamee, A. Norris, Drug repurposing: progress. and challenges and recommendations, Nat. Rev. Drug Discov. 18 (1) (2018) 41–58.

[2] J.H. Beigel, K.M. Tomashek, L.E. Dodd, A.K. Mehta, B.S. Zingman, E. Kalil, H.Y. Chu, A. Luetkemeyer, S. Kline, D. Lopez de Castilla, Remdesivir for the treatment of Covid-19, N. Engl. J. Med. 383 (19) (2020) 1813–1826.

[3] L. Caly, J.D. Druce, M.G. Catton, D.A. Jans, K.M. Wagstaff, The FDA-approved drug ivermectin inhibits the replication of SARS-CoV-2 in vitro, Antivir. Res. 178 (2020) 104787.

[4] T.R. Group, Dexamethasone in hospitalized patients with Covid-19, N. Engl. J. Med. (2020).
[5] M. Zitnik, F. Nguyen, B. Wang, J. Leskovec, A. Goldenberg, M.M. Hoffman, Machine learning for integrating data in biology and medicine: Principles, practice, and opportunities, Inf. Fusion 50 (2019) 71–91.
[6] M. Zitnik, M. Agrawal, J. Leskovec, Modeling polypharmacy side effects with graph convolutional networks, Bioinformatics 34 (13) (2018) i457–i466.
[7] S. Kaliamurthi, G. Selvaraj, C. Selvaraj, S.K. Singh, D.Q. Wei, G.H. Peslherbe, Structure-based virtual screening reveals ibrutinib and zanubrutinib as potential repurposed drugs against COVID-19, Int. J. Mol. Sci. 22 (13) (2021) 7071.
[8] A. Khan, S.S. Ali, M.T. Khan, S. Saleem, A. Ali, M. Suleman, Z. Babar, A. Shafiq, M. Khan, D.Q. Wei, Combined drug repurposing and virtual screening strategies with molecular dynamics simulation identified potent inhibitors for SARS-CoV-2 main protease (3CLpro), J. Biomol. Struct. Dyn. 39 (13) (2021) 4659–4670.
[9] F. Cheng, R.J. Desai, D.E. Handy, R. Wang, S. Schneeweiss, J. Barabási, Network-based approach to prediction and population-based validation of in silico drug repurposing, Nature Commun. 9 (1) (2018) 1–12.
[10] Y. Zhou, Y. Hou, J. Shen, Y. Huang, W. Martin, F. Cheng, Network-based drug repurposing for novel coronavirus 2019-nCoV/SARS-CoV-2, Cell Discov. 6 (1) (2020) 1–18.
[11] D.M. Gysi, I. Do Valle, M. Zitnik, A. Ameli, X. Gan, O. Varol, S.D. Ghiassian, J.J. Patten, R.A. Dave, J. Loscalzo, A.L. Barabási, Network medicine framework for identifying drug-repurposing opportunities for COVID-19, Proc. Natl. Acad. Sci. 118 (19) (2021).
[12] V.N. Ioannidis, D. Zheng, G. Karypis, Few-shot link prediction via graph neural networks for Covid-19 drug-repurposing, 2020, arxiv preprint arxiv:2007.10261.
[13] X. Su, L. Hu, Z. You, P. Hu, L. Wang, B. Zhao, A deep learning method for repurposing antiviral drugs against new viruses via multi-view nonnegative matrix factorization and its application to SARS-CoV-2, Bioinformatics 23 (1) (2022) bbab526.
[14] F. Yang, Q. Zhang, X. Ji, Y. Zhang, W. Li, S. Peng, F. Xue, Machine learning applications in drug repurposing, Interdiscip. Sci.: Comput. Life Sci. (2022) 1–7.
[15] X. Pan, X. Lin, D. Cao, X. Zeng, P.S. Yu, L. He, R. Nussinov, F. Cheng, Deep learning for drug repurposing: Methods. and databases. and and applications., Wiley Interdiscip. Rev.: Comput. Mol. Sci. (2022) e1597.
[16] X. Su, Z. You, L. Wang, L. Hu, L. Wong, B. Ji, B. Zhao, SANE: a sequence combined attentive network embedding model for COVID-19 drug repositioning, Appl. Soft Comput. 111 (107831) (2021).
[17] B.W. Zhao, L. Hu, Z.H. You, L. Wang, X.R. Su, Hingrl: predicting drug–disease associations with graph representation learning on heterogeneous information networks., Brief. Bioinform. 23 (1) (2022) bbab515.
[18] F. Frasca, E. Rossi, D. Eynard, B. Chamberlain, M. Bronstein, F. Monti, SIGN: Scalable inception graph neural networks, 2020, arxiv preprint arXiv:2004.11198.
[19] D.E. Gordon, G.M. Jang, M. Bouhaddou, J. Xu, K. Obernier, K.M. White, M.J. O'Meara, V.V. Rezelj, J.Z. Guo, D.L. Swaney, T.A. Tummino, A SARS-CoV-2 protein interaction map reveals targets for drug repurposing, Nature 583 (2020) 459–468.
[20] S. Povey, R. Lovering, E. Bruford, M. Wright, M. Lush, H. Wain, The HUGO gene nomenclature committee (HGNC), Hum. Genet. 109 (6) (2001) 678–680.
[21] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, in: Proceedings of the International Conference on Learning Representations, Toulon, France, 2017.
[22] W.L. Hamilton, R. Ying, J. Leskovec, Inductive representation learning on large graphs, in: Advances in Neural Information Processing Systems, California, United States, 2017.
[23] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, Graph attention networks, in: Proceedings of the International Conference on Learning Representations, Vancouver, Canada, 2018.
[24] V.N. Ioannidis, X. Song, S. Manchanda, M. Li, X. Pan, D. Zheng, X. Ning, X. Zeng, G. Karypis, DRKG - drug repurposing knowledge graph for Covid-19, 2020, https://github.com/gnn4dr/DRKG/.
[25] D.S. Wishart, Y.D. Feunang, A.C. Guo, E.J. Lo, A. Marcu, J.R. Grant, D. Sajed, C. Li, Z. Sayeeda, N. Assempour, Drugbank 5.0: a major update to the drugbank database for 2018, Nucleic Acids Res. 46 (D1) (2017) D1074–D1082.
[26] D.S. Himmelstein, A. Lizee, C. Hessle, L. Brueggeman, S.L. Chen, A. Hadley, P. Khankhanian, S.E. Baranzini, Systematic integration of biomedical knowledge prioritizes drugs for repurposing, Elife 6 (2017) e26726.
[27] B. Percha, R.B. Altman, A global network of biomedical relationships derived from text, Bioinformatics 34 (15) (2018) 2614–2624.
[28] D. Szklarczyk, A. Gable, D. Lyon, A. Junge, S. Wyder, M. Huerta-Cepas, N.T. Doncheva, J.H. Morris, P. Bork, L.J. Jensen, STRING v11: protein–protein association networks with increased coverage. and supporting functional discovery in genome-wide experimental datasets, Nucleic Acids Res. 47 (D1) (2019) D607–D613.
[29] S. Orchard, M. Ammari, B. Aranda, L. Breuza, L. Briganti, F. Broackes-Carter, N.H. Campbell, G. Chavali, C. Chen, N. Del-Toro, M. Duesbury, The mIntAct project—IntAct as a common curation platform for 11 molecular interaction databases, Nucleic Acids Res. 42 (D1) (2014) D358–D363.
[30] C.K. C., W.A. H., F.Y. Y., S. Kiwala, A.C. Coffman, G. Spies, A. Wollam, S.N. C., G.O. L., G. M., GIdb 3.0: a redesign and expansion of the drug–gene interaction database, Nucleic Acids Res. 46 (D1) (2018) D1068–D1073.
[31] Z. Wang, M. Zhou, C. Arnold, Toward heterogeneous information fusion: bipartite graph convolutional networks for in silico drug repurposing, Bioinformatics 36 (Supplement1) (2020) i525–33.
[32] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, S.Y. Philip, A comprehensive survey on graph neural networks, IEEE Trans. Neural Netw. Learn. Syst. 32 (1) (2021) 4–24.