Knowledge-Based Systems 283 (2024) 111187

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

SLGCN: Structure-enhanced line graph convolutional network for predicting drug–disease associations

Bao-Min Liu [a], Ying-Lian Gao [b,*], Feng Li [a], Chun-Hou Zheng [a], Jin-Xing Liu [a]

[a] School of Computer Science, Qufu Normal University, Rizhao, 276826, Shandong, China
[b] Qufu Normal University Library, Qufu Normal University, Rizhao, 276826, Shandong, China

ARTICLE INFO

Keywords:
Drug–disease association prediction
Graph convolutional network
Line graph
Subgraph

ABSTRACT

Drug repositioning is a rapidly growing strategy in drug discovery, as the time and cost needed are considerably less compared to developing new drugs. In addition to traditional wet experiments, designing effective computational methods to discover potential drug–disease associations is an attractive shortcut in drug repositioning. Most current methods based on graph neural networks ignore the heterophily of the constructed drug–disease network, resulting in inefficient predictions. In this paper, a novel structure-enhanced line graph convolutional network (SLGCN) is proposed to learn comprehensive representations of drug–disease pairs, incorporating structural information to conduct heterophily. First, line graphs centered around drug–disease pairs are extracted. This process turns the association prediction task into a node classification problem, which better displays the learning ability of SLGCN. Then, in message aggregation, a relation matrix is proposed to mark the structural importance of neighboring nodes. In this way, messages from nodes with lower structural importance can be assigned small weights. Unlike vanilla GCN, which adds self-loops to average ego representations and aggregated messages, an update gate is proposed to integrate biology information contained in ego representations with topology information contained in aggregated messages. Extensive experiments show that SLGCN achieves better performance than other advanced methods among the two datasets.

1. Introduction

Drug repositioning [1] is defined as finding new indications for drugs already on the market. Over the past decades, numerous cases, such as thalidomide for treating the complications of leprosy [2] and multiple myeloma [3], have proven to have significant clinical implications. In addition, drug repositioning has enormous potential in the economy. It can save 85% of costs compared with bringing a new drug to the market, which has a higher risk of failure [4]. According to the study [5], repositioned drugs produced 25% of the annual income for the pharmaceutical industry. Owing to these advantages, drug repositioning has been a hot topic in drug development. Predicting drug–disease associations can help identify previously unknown drug applications in real situations, potentially saving a large amount of resources and thus accelerating the drug repositioning pipeline based on wet-lab experiments. The use of computational methods to find candidate drugs from public data further promotes traditional drug repositioning. Since the outbreak of COVID-19, computational methods have been demonstrated to be powerful alternatives for discovering potential therapeutics. Many potential drugs discovered by computational methods have entered clinical trials [6,7].

Generally, current computational methods infer potential drug–disease associations based on the drug–drug similarity matrix calculated from their chemical structures, the disease–disease similarity matrix calculated from disease phenotypes, and known drug–disease associations verified by wet-lab experiments. On the one hand, the similarity information provides informative biological features of drugs and diseases, which are beneficial for model interpretability. On the other hand, obtaining more verified drug–disease associations is challenging. Therefore, earlier prediction models introduced multiple types of similarity and explored effective methods to integrate these similarities. For example, drug–drug similarities calculated from target protein domains, target protein-encoding gene ontology terms and side effects are introduced in some works [8] to improve the performance. Another similarity between diseases can be calculated based on their related genes, known as functional similarity. Furthermore, novel similarities

* Corresponding author.
E-mail addresses: alllbm@163.com (B.-M. Liu), yinliangao@126.com (Y.-L. Gao), lifeng_10_28@163.com (F. Li), zhengch99@126.com (C.-H. Zheng), sdcavell@126.com (J.-X. Liu).

https://doi.org/10.1016/j.knosys.2023.111187
Received 12 September 2023; Received in revised form 5 November 2023; Accepted 8 November 2023
Available online 11 November 2023

mined from known associations have been proposed using different methods, such as the Gaussian interaction profile (GIP) kernel [9] and linear neighborhood similarity (LNS) [10].

After obtaining diverse similarities, adopting a prior strategy to deeply integrate them in prediction is essential. For instance, Liang et al. [11] imposed Laplacian regularized sparse subspace learning (LRSSL) to combine these diverse features. Zhang et al. [8] proposed a novel approach based on a Bayesian inductive matrix called DRIMC. To capture the complementary information from different similarities, the similarity fusion method [12] is applied to embed different similarities into a drug (disease) feature matrix. Additionally, Xuan et al. [13] introduced two terms that independently measure the deviation between the expected drug (disease) similarities and the actual ones in nonnegative matrix factorization.

The aforementioned models are mainly based on matrix methods which are suitable for merging different types of similarities [14]. However, collecting the related data to measure these similarities is time-consuming. Therefore, with limited similarity information, another type of method constructs a heterogeneous network or a bipartite network to capture topological information. During the process of inferring potential associations, bi-random walk [15] and label propagation [10,16] are often utilized on the constructed network.

With the rapid surge of biological data [17,18] and significant development of computing power [19], the superiority of deep learning has been fully affirmed in computational biology [20]. Graph neural networks (GNNs) [21], powerful deep learning methods for learning graph data, have shown convincing performances in predicting unobserved associations [22,23]. For example, Yu et al. [24] built a novel layer attention graph convolutional network called LAGCN. First, the adjacency matrix of the heterogeneous network comprises the drug–drug similarity matrix, the disease–disease similarity matrix and the known drug–disease association matrix. Then, a three-layer architecture of vanilla GCN is used to learn node embedding from the constructed heterogeneous network, and the attention mechanism [25] is introduced as an interlayer combination which integrates the node features containing the information from different hop neighbors. To filter noisy information, Meng et al. [26] selected only the top-k similar nodes to construct similarity networks and designed a novel weighted bilinear graph convolution operation to encode weighted interaction information between drug (disease) neighbors.

Most existing models based on GNNs merely focus on learning drug/disease node embeddings from the constructed network, and the probability of the predicted association is measured by two related node features. However, these methods heavily rely on the labels of associations in the training process. Thus, the capacity of modeling associations is easily influenced by the high sparsity of drug–disease associations. To overcome this challenge, SEAL (learning from Subgraphs, Embeddings and Attributes for Link prediction) [27], a link prediction architecture based on GNNs which learns a function mapping the subgraph patterns to link existence, is adopted in our model. Similarly, Sun et al. [28] proposed a drug repositioning method called PSGCN. This model operates on partner-specific subgraphs extracted from the drug–disease association network. PSGCN utilizes a vanilla GCN and a layer self-attention mechanism to learn node representations within the subgraphs. To obtain subgraph-level representations for prediction, a sort pooling layer is employed as a graph classification task. However, the needed pooling operations may result in the loss of semantic information.

To mitigate the information loss caused by pooling operations in subgraph-based methods, a transformation [29] from subgraphs to corresponding line graphs is utilized after subgraph extraction. In the line graph, the target predicted drug–disease pair is converted to the predicted node, and other edges in the original subgraph are also converted to nodes. Owing to the transformation, GNNs can be applied to the corresponding line graph to learn the embeddings of the target drug–disease pair.

While many models based on GNNs have achieved the goal of capturing the structural features from the whole graph or subgraphs, most models ignore the heterophily of the constructed graph. In the line graph, some nodes with opposite labels may directly connect with the predicted node. The common averaging strategy in vanilla GCN, which equally encourages messages from these nodes in message aggregation, may not work well in this case. In addition, a typical GCN directly mixes up ego representations containing biology information and aggregated messages containing topology information in the update function by adding self-loops. This may lead to increasingly indistinguishable node representations, as aggregated messages may involve information from the opposite label. Moreover, this mixing strategy does not fully leverage the rich biological knowledge present in ego representations.

To address these issues, a structure-enhanced line graph convolution network (SLGCN) is proposed. Aiming to the problem in message aggregation, an additional operation should be imposed to reduce the impact of messages from the opposite label. The distance information from other nodes to target nodes, forming the predicted associations, can help discover these nodes. Some of them are distant from the target nodes, implicitly suggesting that they are less important to the predicted associations in the graph structure. Therefore, a relation matrix representing the structural importance of nodes is constructed to assign proper weights to neighboring nodes in message aggregation, distinguishing beneficial messages from more important nodes in graph structure. The incorporation of the relation matrix improves the feature extraction capability. Self-loops are detached from the adjacency matrices of line graphs. Instead, an update gate is designed to control the outflow of ego representations and aggregated messages. This mechanism can ensure the vital role of the pure biology information in learned representations, thus avoiding the generation of distinguishable node representations. The following experiments demonstrate that SLGCN outperforms the compared models. More importantly, case studies on Alzheimer's disease and breast cancer show that SLGCN can mine potential associations that are unknown in the datasets but supported by public databases and literature. Our contributions are as follows:

1. To reduce the information loss and fulfill the GCN's ability to learn node embeddings, a transformation to the line graph is employed to capture graph structural features.
2. To aggregate more beneficial and important messages, a relation matrix is adaptively learned from the distance information to assign proper weights for neighboring nodes in message aggregation.
3. A gated update function is proposed to combine ego representations and aggregated messages, which fully leverages the biology information to a great extent and further enhances the ability of SLGCN.

The remaining sections of this paper are organized as follows. In Section 2, the construction of the line graphs and the architecture of SLGCN are described in detail. The favorable results of SLGCN are demonstrated in Section 3. Case studies of Alzheimer's disease and breast cancer are shown in Section 4. This paper is concluded in Section 5.

2. Methods

In this section, the construction of line graphs is introduced first. In the construction process, the heterogeneous network is first obtained, composed of three types of edges, drug–drug similarities, disease–disease similarities and known drug–disease associations, as well as two types of nodes, including drug and disease. Then, centered around each drug–disease pair, enclosing subgraphs are extracted. To directly learn features describing drug–disease pairs, enclosing subgraphs are converted into line graphs. The construction process is illustrated in Fig. 1. Subsequently, the learning process of SLGCN on line graphs

is introduced. The learning process is depicted in Fig. 2. The line graph shown in Fig. 1(right) is the same as the line graph depicted in Fig. 2(left). The two figures are sequential.

2.1. Construction of line graphs

2.1.1. Construction of heterogeneous network

The heterogeneous network G_H consists of the drug–drug similarity network, disease–disease similarity network and known drug–disease association network.

$A_{\mathrm{dr}} \in R^{n_1 \times n_1}$ and $A_{\mathrm{di}} \in R^{n_2 \times n_2}$ denote the adjacency matrices of the drug–drug similarity network and the disease–disease similarity network, respectively. n_1 is the number of drugs, and n_2 is the number of diseases. Although more edges in the similarity network may provide more potential topological features, we choose only the most similar nodes to construct the similarity networks. In this way, the scale of the heterogeneous network is smallest, which reduces the time and memory cost in extracting enclosing subgraphs and cuts off noisy information to a great extent. When the element (i, j) of A_{dr} is 1, the similarity score at the corresponding position is highest in the i-th row of S_{dr}, the drug–drug similarity matrix. Otherwise, $(i, j) = 0$. A_{di} is constructed from the disease–disease similarity matrix S_{di} in the same way.

Different from the similarity networks that are partly constructed from the original similarity matrices, the drug–disease association network is fully based on the known association matrix $A \in R^{n_1 \times n_2}$, which also represents the adjacency matrix of the drug–disease association network. $A_{ij} = 1$ denotes that drug node dr_i is associated with disease node ds_j. The known associations in A are verified by public data and are much less than unknown associations.

After obtaining the three networks, the adjacency matrix $A_H \in R^{(n_1+n_2)\times(n_1+n_2)}$ of the constructed heterogeneous network G_H is denoted in Eq. (1):

$$A_H = \begin{bmatrix} A_{\mathrm{dr}} & A \\ A^{\mathrm{T}} & A_{\mathrm{di}} \end{bmatrix} \in R^{(n_1+n_2)\times(n_1+n_2)}. \tag{1}$$

The initial feature matrix of the heterogeneous network is initialized as follows:

$$X^{(0)} = \begin{bmatrix} \mu \sim S_{\mathrm{dr}} & A \\ A^{\mathrm{T}} & \mu \sim S_{\mathrm{di}} \end{bmatrix} \in R^{(n_1+n_2)\times(n_1+n_2)}. \tag{2}$$

where μ is a penalty factor [24] controlling the contribution of similarity information, which contains diverse drug chemical structure information and disease phenotype information. The penalty factor helps this essential biological information make a larger difference. In addition, $\sim S_{\mathrm{dr}} = D_{dr}^{-1/2} S_{\mathrm{dr}} D_{dr}^{-1/2}$, where $D_{dr} = diag(\sum_j S_{dr_{ij}})$, denotes the normalization for S_{dr}. The same procedure is performed on S_{di}. The initial node attributes are composed of the similarity values and the known associations, describing nodes from the biology perspective.

2.1.2. Enclosing subgraphs extraction

Apart from the biological feature, leveraging the graph structure features of the associations is key to accurately predicting the existence of associations. Effective models are supposed to learn graph features composed of low- and high-order information. Studies [30,31] demonstrate that models that can learn high-order information tend to have better performance. However, obtaining high-order information from the whole graph is often a costly and time-consuming process. Without capturing high-order information by stacking layers of GCN on the whole graph, h-hop enclosing subgraphs centered around predicted associations are built to approximate high-order information. The feasibility can be proven by a γ-decaying heuristic [27]. The analysis of the γ-decaying heuristic has revealed that local subgraphs contain enough information to extract conducive graph features, and high-order information can be approximated from the local subgraphs.

To learn the latent structure features of the predicted association between target node v_i and target node v_j, the h-hop subgraph is extracted as follows:

$$G^h_{(v_i,v_j)} = \{v | d(v, v_i) \le h \text{ or } d(v, v_j) \le h\}, \tag{3}$$

where $d(v, v_j)$ represents the distance of the shortest path between v_j and v. Likewise, $d(v, v_i)$ is the shortest distance between v_i and v. Each predicted drug–disease association is represented by an enclosing subgraph around itself.

2.1.3. From enclosing subgraphs to line graphs

After constructing h-hop enclosing subgraphs, graph learning algorithms can be directly applied to the enclosing subgraphs to learn the latent feature vector of the associations. However, this strategy requires graph pooling layers to preserve the same dimension of feature vectors due to the different sizes of enclosing subgraphs. The introduction of pooling operations certainly results in graph information loss. In addition, GCN is more effective in node feature learning than edge feature learning [29]. Transferring the original subgraphs to line graphs can address these issues. The process of transformation is shown in Fig. 1.

In the line graph, a node represents an edge of the original subgraph. The existence of an edge between two nodes in the line graph depends on whether they have the same nodes in the original graph. For example, drug node v_1 is associated with drug node v_6 in the original graph. Therefore, the line graph has a node $e_{1,6}$. Meanwhile, node $e_{1,6}$ links to node $e_{1,2}$ and node $e_{4,6}$ in the line graph, for which they include the same nodes v_1 and v_6, respectively. The transformation preserves all nodes and edges of subgraphs and realizes that features of predicted associations can be directly updated with adjacent edge features. Then, we can apply SLGCN on corresponding line graphs to learn the embeddings of target associations.

Note that we will use 'predicted nodes' in the line graphs to replace 'predicted associations' of the original enclosing subgraphs in the following. 'Target nodes' denotes nodes that form the predicted associations in the enclosing subgraphs.

2.2. Structure-enhanced line graph convolutional network

2.2.1. Overview

General GCN architectures contain two components: message aggregation from neighboring nodes and an update function to combine the derived messages with the initial node features.

In message aggregation, averaging messages from connected neighbors is a commonly used strategy, although it may not be suitable for all types of graphs. In the transformed line graph of Fig. 1, predicted node $e_{1,2}$ has three 1-hop neighbors, $e_{1,3}$, $e_{1,6}$, and $e_{2,4}$. Conventionally, the messages from these neighbors share the same weight in the aggregation process, meaning they have no difference to $e_{1,2}$. However, if the true label of $e_{1,2}$ is 0, the message from $e_{1,3}$ having the opposite label will mislead the representation learning of $e_{1,2}$. As shown in the enclosing graph of Fig. 1, although v_3 is directly connected to v_1, it has a long distance to v_2. This may suggest that $e_{1,3}$ converted from v_1 and v_3 has low importance to the predicted node $e_{1,2}$ in the graph structure. Hence, we hope to lessen the contribution of these messages that are from neighboring nodes but are less beneficial to predicted nodes based on the distance information of the enclosing graphs. To achieve this, a labeling function based on the distance information is adopted to mark the structural differences of neighboring nodes to the predicted nodes. Through further learning on the assigned node labels, a relation matrix is constructed to describe the structural importance to the predicted node on the line graph. Then, the constructed relation matrix is introduced in message aggregation to reveal more important and beneficial messages from the perspective of topology. After aggregating messages from neighbors, how to combine the aggregated messages with its ego representation is another aspect to boost

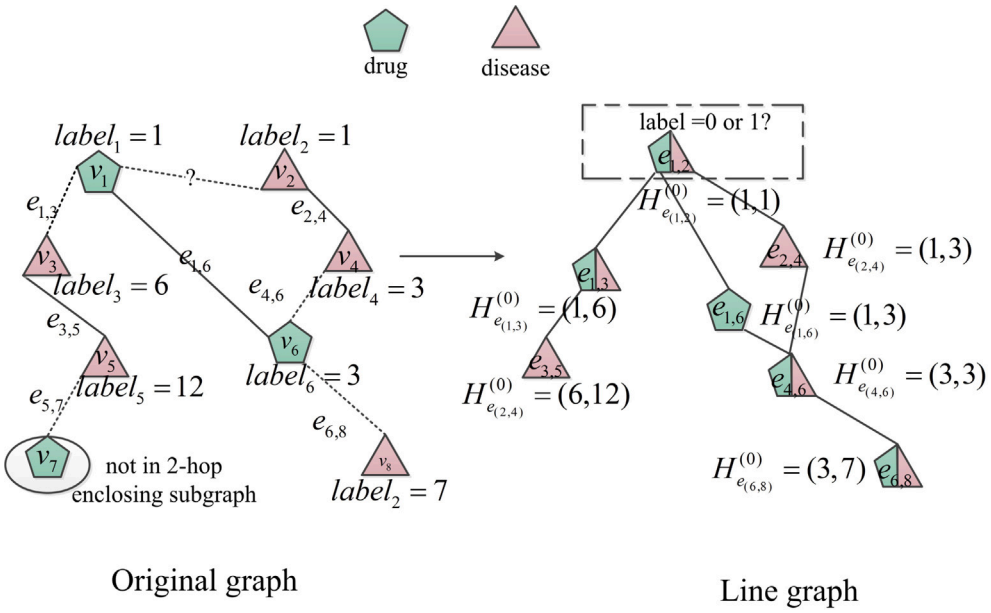

Fig. 1. The construction of line graphs.

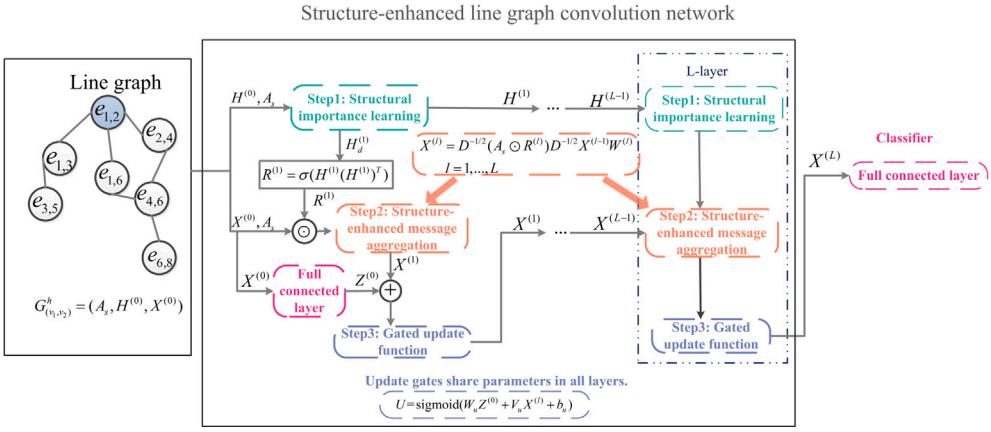

Fig. 2. The architecture of SLGCN when predicting the label of association $e_{1,2}$.

the learning ability. Especially in drug–disease association prediction, ego representations contain rich biological information from chemical structures and disease phenotypes. Prior knowledge can provide a more meaningful interpretation of association existence. Rather than adding the self-loop to average ego representations and aggregated neighbor messages in GCN, we detach the self-loop in the adjacency matrix and alter a gated update function whose parameters are shared by all layers to integrate aggregated messages containing structure information and its ego representations containing fruitful biology information.

The architecture of SLGCN is illustrated in Fig. 2. It consists of four steps: i) structural importance learning, which transforms initial node labels representing topological features into structural embeddings; ii) structure-enhanced message aggregation, which aggregates weighted neighboring messages using the constructed relation matrix; iii) gate update function, which combines the aggregated messages and initial node biological features; and iv) classifier, which predicts the label of the predicted node.

2.2.2. Structural importance learning

The distance between two nodes partly implicates the correlation level. Therefore, we hope that the closer nodes in the graph structure can provide more information for predicted associations in learning features. Hence, we expect to learn the structural importance information in line graphs mapped by the distance between nodes in the original enclosing subgraphs. However, it is a time-consuming operation to calculate the distances of all node pairs in a graph. To overcome this challenge, only the distances between each node and the target nodes are calculated in the enclosing subgraphs. In this way, the complexity of our model is significantly lessened.

Based on this requirement, a labeling function [27], called double-radius node labeling (DRNL), is adopted to mark the different structural importance to the target nodes based on the distance information. DRNL assigns nodes with larger labels to nodes with a larger radius, i.e., the distance to the target nodes. The equation is as follows:

$$label_i = 1 + \min(d(v_i, v_1), d(v_i, v_2)) + (d_s/2)[(d_s/2) + (d_s\%2) - 1]. \quad (4)$$

Here, $d_s = d(v_i, v_1) + d(v_i, v_2)$, in which v_1 and v_2 represent the target nodes in the enclosing subgraph. $d_s/2$ is the integer quotient of d_s divided by 2. $d_s\%2$ is the corresponding remainder. The labels of the target nodes are 1, and the labels of the nodes that satisfy $d(v, v_1) = \infty$ or $d(v, v_2) = \infty$ are assigned as 0. The labels are represented by one-hot encoding.

After transferring to the line graphs, the labels of node $e_{i,j}$ are represented as:

$$H^{(0)}_{e_{i,j}} = \text{concat}(label_i, label_j). \quad (5)$$

The labels only provide the structural importance features between each node and target nodes. To capture more comprehensive structural importance information, the labels are fed to GraphSAGE [32] for representation learning. Note that GraphSAGE can be replaced by other methods. We empirically verified its better performance than multilayer perceptron (MLP) and vanilla GCN. The learning process is denoted as follows:

$$H_v^{(l)} \leftarrow w_1 H_v^{(l-1)} + w_2 \text{mean}_{v' \in N(v)} x_{v'}, \forall v \in G^h, \tag{6}$$

where w_1 and w_2 are trainable matrices and $N(v)$ denotes the adjacent nodes of node v in the line graph. The learned structural importance information is expected to better guide the message aggregation process, in which closer nodes can provide more beneficial and important information for the predicted nodes in line graphs. Therefore, a relation matrix that has the same shape as the adjacency matrix is constructed as follows:

$$R^{(l)} = \text{sigmoid}(H^{(l)} H^{(l)^T}), \tag{7}$$

where $H^{(l)}$ is the learned feature matrix, in which $H_v^{(l)}$ in Eq. (6) represents the derived feature vector of node v in the line graph, a row of the relation matrix $R^{(l)}$.

2.2.3. Structure-enhanced message aggregation

Due to the heterophily of the constructed line graph, some neighbor messages might impact the representation learning of the predicted node. Therefore, how to identify these irrelevant neighboring nodes and reduce the negative impact is essential in message aggregation. As the analysis in overview, we determine that the distances to the target nodes can provide some structural importance information to the predicted node in the transformed line graph. Therefore, the relation matrix based on the distance information is constructed to represent the structural importance of neighboring nodes to the predicted node. Then, the adaptively learned relation matrix is introduced in the message aggregation process, which assigns proper weights to neighboring nodes to help distinguish more useful neighbor messages.

Owing to the transformation to line graphs, node representations of line graphs should include two nodes representations of enclosing subgraphs. Thus, a concatenation operation is used to obtain new node representations of line graphs. The initial input of SLGCN is denoted as:

$$x_{e_{i,j}}^{(0)} = \text{concat}(x_i^{(0)}, x_j^{(0)}). \tag{8}$$

For convenience, $X^{(0)}$ is still used to represent the initial node embeddings in line graphs. The aim of introducing the relation matrix is to identify more important neighboring nodes from the graph structure. Therefore, the message aggregation process of SLGCN is given as follows:

$$X^{(l)} = D^{-1/2}(A_s \odot R^{(l)}) D^{-1/2} X^{(l-1)} W^{(l)}, l = 1, \ldots, L. \tag{9}$$

where D is the diagonal degree matrix, $W^{(l)}$ is the trainable matrix, and $\odot$ denotes the elementwise product. The relation matrix assigns proper weights for neighboring nodes in message aggregation, which can be regarded as an attention mechanism. Different from most attention mechanisms from the attribute perspective [33,34], the proposed relation matrix focuses on the distance information of the graph, which is based on the topology of the enclosing graphs. Therefore, introducing the relation matrix can enhance the ability to discover more structurally important nodes from the topology perspective, not from the attribute perspective. In addition, the attention mechanism from the attribute perspective is easily affected by increasingly similar node embeddings within increasing layers of the GCN. This will obviously reduce the recognition capability of the attention mechanisms. In contrast, the relation matrix constructed from the structure information can still work well in this case.

2.2.4. Gated update function

Different from social networks and networks in other fields, ego representations mined from prior biology knowledge are important to the interpretability of models. Combining the latent features and graph structures can improve the performance [27]. The typical GCN mixes up ego representations and aggregates neighbor representations with self-loops. However, the mixing strategy poses severe challenges. On the one hand, in the case of heterophily such as the analysis in overview, predicted nodes may aggregate messages from neighboring nodes attributed to the opposite label. Thus, node representations of different labels will be similar through mixing them, resulting in the impossibility of distinguishing neighbors having the opposite label. On the other hand, directly mixing up ego representation from the biology perspective and aggregated messages from the topology perspective will destroy the pure biology information in the case of oversmoothing. With the increasing layers of the GCN, the larger receptive field will make the learned node representations indistinguishable, severely impacting the performance of the models. Even with the residual connection, the indistinguishable representations will influence the pure and useful biology information contained in ego representation and reduce the ability of models.

To address these problems, we generate a gated update function to integrate biological features and structural features. The gated update function adaptively controls the outflow of the integration of ego representation and aggregated messages through weights in the learned update gate. This mechanism can effectively alleviate the aforementioned problems. Assuming the model suffers from the oversmoothing issue, the gate will control the output of the current layer. In the next layer, aggregated messages might be influenced by the input. However, in the gated update function, ego representations will take a key role, reducing the negative influence on the pure biology information. The useful biology information contained in ego representations will ensure that the node representations are distinguishable.

First, ego representations are obtained by a linear transformation as follows:

$$Z^{(0)} = X^{(0)} W' + b', \tag{10}$$

where W' and b' are trainable parameters. Then, after obtaining aggregated messages $X^{(l)}$ of the lth layer, the update process is denoted as follows:

$$U = \text{sigmoid}(W_u Z^{(0)} + V_u X^{(l)} + b_u), \tag{11}$$

$$X^{(l)} = U \odot \tanh(Z^{(0)} + X^{(l)}), l = 1, \ldots L. \tag{12}$$

U represents the update gate, where W_u and V_u are trainable matrices and b_u is the trainable bias. The elementwise production exploits U as a filter to extract necessary information captured in each layer. The final node embeddings X_e^L of the predicted associations e are fed to a fully connected layer (FCL) for predictions. The whole process of SLGCN is listed in Algorithm 1.

2.2.5. Optimization

The cross-entropy function and Adam optimizer [35] are used to optimize the parameters in our model. The loss function is denoted as follows:

$$loss = -\sum_{e \in L} (y_e \log(p_e) + (1 - y_e) \log(1 - p_e)), \tag{13}$$

where y_e is the true label of the predicted association e and p_e is the predicted score of the drug–disease pair e. In addition, regular dropout [36] is used to improve the generalization ability.

The code is at https://github.com/bdtree/SLGCN.

Table 1
Details of datasets.

Dataset	Drug	Disease	Known association
Fdataset	593	313	1933
Cdataset	663	409	2352

Algorithm 1: SLGCN

Input: adjacency matrix of line graph A_s; node features $X^{(0)}$; node labels $H^{(0)}$

Output: prediction score p

$Z^{(0)} = X^{(0)}W' + b'$;//obtaining nodes' ego representations by a linear transformation

for $l = 1, \ldots, L$ **do**

/* Structure importance learning */

$H_v^{(l)} = w_1 H_v^{(l-1)} + w_2 \text{mean}_{v' \in N(v)} x_{v'}, \forall v \in G^h$;

/* Obtaining relation matrix */

$R^{(l)} = \text{sigmoid}(H^{(l)} H^{(l)^T})$;

/* Structure-enhanced message aggregation */

$A_{RS} = A_s \odot R^{(l)}$;//obtaining weighted adjacency matrix

$A_{NRS} = D^{-1/2} A_{RS} D^{-1/2}$;//obtaining normalized adjacency matrix

$X^{(l)} = A_{NRS} X^{(l-1)} W^{(l)}$;//obtaining aggregated messages

/* Gated update function */

$U = \text{sigmoid}(W_u Z^{(0)} + V_u X^{(l)} + b_u)$;//obtaining the update gate

$X^{(l)} = U \odot \tanh(Z^{(0)} + X^{(l)})$;//combining aggregated messages //with ego representations

end

/* Classification */

$p = Classifier(X_e^L)$//predicting the label of predicted drug–disease pair //converted to node e in line graph

3. Results and discussion

3.1. Datasets

In this work, two datasets (Fdataset [37] and Cdataset [15]) are used to evaluate the results of the models. The details of the two datasets are listed in Table 1. The datasets separately include three matrices describing the similarity data and the association data.

$S_{\text{dr}} \in R^{n_1 \times n_1}$ denotes the drug–drug similarity matrix obtained from the chemical structure of drugs. First, SMILES (simplified molecular input line entry system) representing drug molecules [38] are collected from DrugBank [39]. Then, SMILES are processed to obtain the fingerprints by the Chemical Development Kit (CDK) [40]. The final similarity score is the Tanimoto score of the fingerprints. The disease–disease similarity matrix $S_{\text{di}} \in R^{n_2 \times n_2}$ is derived from MimMiner [41], which calculates similarity using the disease phenotypic data from the public database OMIM [42]. The drug–disease association matrix $A \in R^{n_1 \times n_2}$ is collected from CTD [43]. n_1 and n_2 denote the number of drugs and diseases, respectively.

3.2. Experimental setup

Cross-validation is widely applied to prove the efficiency of models. Therefore, 10-fold cross-validation (10-CV) was adopted in the experiments to evaluate the performance. Due to the imbalance of negative samples and positive samples, we adopted a balanced sample strategy and an unbalanced sample strategy for comparison. All positive samples were used in these strategies. In the balanced strategy, the same number of negative samples are randomly selected in the experiments. In the unbalanced strategy, the number of negative samples is twice as high as the number of positive samples. After extracting negative samples, the positive samples and the negative samples were divided into 10 subsets, respectively. In each fold, one subset of positive samples and one subset of negative samples were used for testing, and the remaining subsets were used as training data.

Meanwhile, two metrics, the area under the receiver operating characteristic curve (AUROC) and the area under the precision recall curve (AUPR), were adopted to evaluate the performances of different models. The ROC curve was plotted by the true positive rate (TPR) and false positive rate (FPR). The equations of TPR and FPR are denoted as:

$$TPR = recall = \frac{TP}{TP + FN}, \tag{14}$$

$$FPR = \frac{FP}{TN + FP}. \tag{15}$$

According to the predicted labels and the true labels, all samples were split into four types. TP and TN denote the number of correctly identified positive samples and the number of correctly identified negative samples, respectively. For those negative (positive) samples but predicted as positive (negative), we call them false positives (negatives). FP and FN separately represent the number of them. Likewise, the PR curve was plotted by recall and precision. The calculation methods of recall and precision are shown in Eqs. (14) and (16), respectively.

$$precision = \frac{TP}{TP + FP}. \tag{16}$$

In Fig. 3, the number of layers and hidden units are searched within the ranges of $\{1, 2, 3, 4\}$ and $\{32, 64, 128, 256\}$, respectively. As observed, SLGCN adopts a three-layer architecture with 128 hidden units to obtain the best performances. The selection of the learning rate and dropout rate is illustrated in Fig. 4. The learning rate is chosen from $\{2e{-}5, 2e{-}4, 2e{-}3, 2e{-}2\}$, and the dropout rate is chosen from $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6\}$. In terms of AUROC, SLGCN performs best on the Fdataset and Cdataset when the learning rate is $2e{-}4$ and the dropout rate is 0.4. Then, the parameter sensitivity of the penalty factor μ chosen from $\{2, 4, 6, 8, 10\}$ in Eq. (2) is analyzed in Fig. 5. The case without the penalty factor, namely, $\mu = 1$, is also compared. As observed, μ is set as 2 on the Fdataset and Cdataset. This indicates that scaling the similarity information to a suitable value is rewarding for subsequent feature learning. The hop of the enclosing subgraphs is analyzed in the next section.

3.3. Analysis of hop

The hop of the enclosing subgraphs is crucial in SLGCN, which influences the scale of the subgraphs. According to the γ-decaying heuristic in the SEAL framework, first-order and second-order information is enough to approximate the higher-order information. The comprehensive proof of the γ-decaying heuristic can be found in SEAL [27]. The same set is adopted in our paper. Therefore, h is selected from $\{1, 2\}$. For the comprehensiveness of our experiment, the case under $h = 3$ is also tested. As shown in Fig. 6, SLGCN under $h = 2$ performs better than $h = 1$ and $h = 3$, despite 3-hop enclosing subgraphs containing more nodes and edges. It is worth noting that constructing the subgraphs is expensive. The complexity for constructing subgraphs is $O(deg^h)$, where deg is the degree of every node in the subgraph. Considering the time consumption and satisfactory performance, $h = 2$ is the best choice.

3.4. Comparison with state-of-the-art models

To assess the better ability of our model in predicting unknown associations, seven models and SLGCN are comprehensively compared in this section. The compared models are briefly presented as follows:

- DRRS [44] performs the low-rank matrix completion algorithm on the adjacency matrix of the heterogeneous network.
- SCPMF [45] applies probabilistic matrix factorization, in which two types of similarities are introduced as constraints.

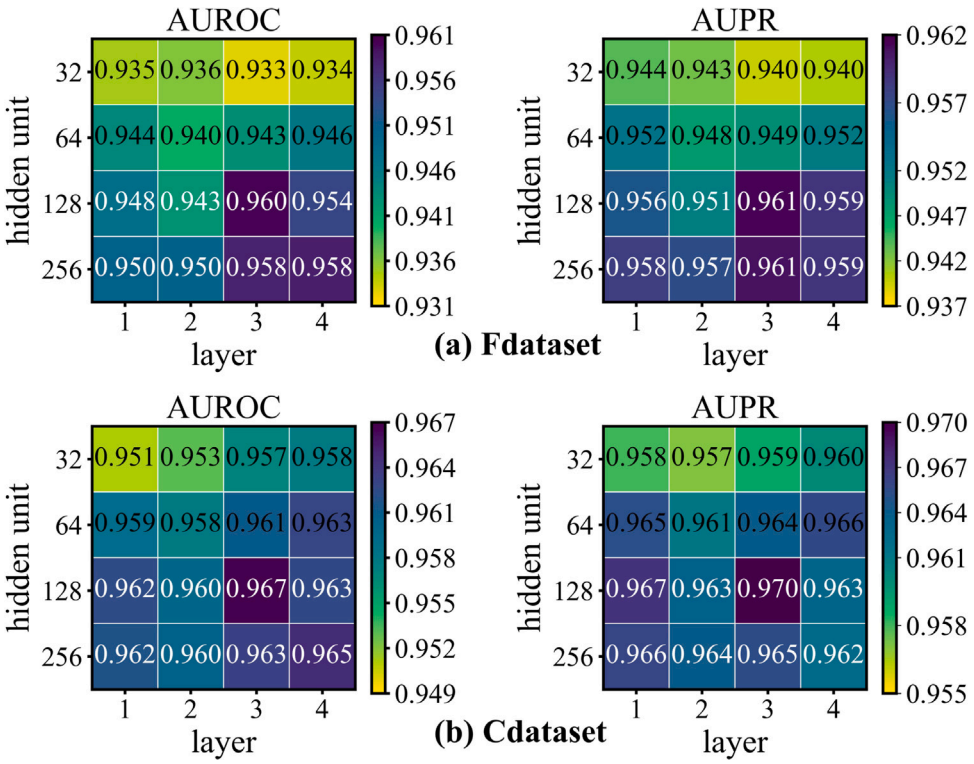

Fig. 3. Parameter sensitivity of the number of layers and hidden units on the Fdataset (a) and Cdataset (b).

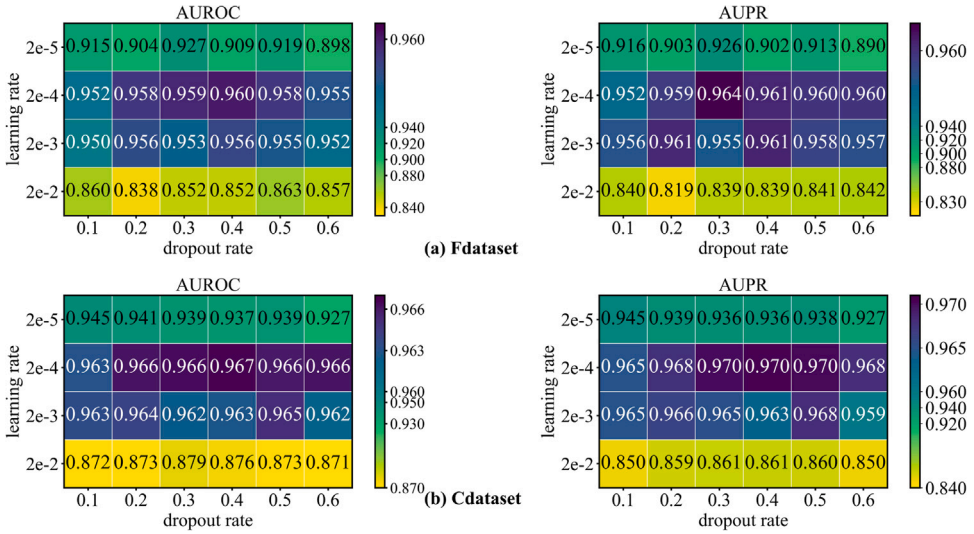

Fig. 4. Parameter sensitivity of learning rate and dropout rate on the Fdataset (a) and Cdataset (b).

- MBiRW [15] designs new similarity measures to augment prior drug and disease knowledge. In final predictions, a bi-random walk algorithm is implemented to calculate the probability of drug–disease associations.
- LAGCN [24] introduces a layer attention mechanism in a three-layer architecture GCN model. The final node embeddings are the fusion of the output of three GCN layers. Then, a bilinear decoder is used to predict the score of associations.
- DRWBNCF [26] proposes a weighted bilinear operation to aggregate all weighted information between neighbors. The final prediction relies on the multilayer perceptron.
- PSGCN [28] designs a partner-specific approach based on a GCN and a layer self-attention mechanism to capture multiscale layer information.
- GLGMPNN [46] conducts different message passing neural networks (MPNNs) on the similarity networks and association network. A gated mechanism is introduced to integrate these features.

The parameters of all compared models are chosen as their optimal values according to their papers.

The results of all models are listed in Table 2. Obviously, SLGCN outperforms the other models in terms of AUROC and AUPR. On the Fdataset, SLGCN achieves AUROC and AUPR scores of 0.960 and 0.961, respectively. Compared with the second-best AUROC scores obtained by GLGMPNN (0.944, 0.942) in the balanced and unbalanced settings, SLGCN is 1.6% and 2% higher, respectively. GLGMPNN uses different

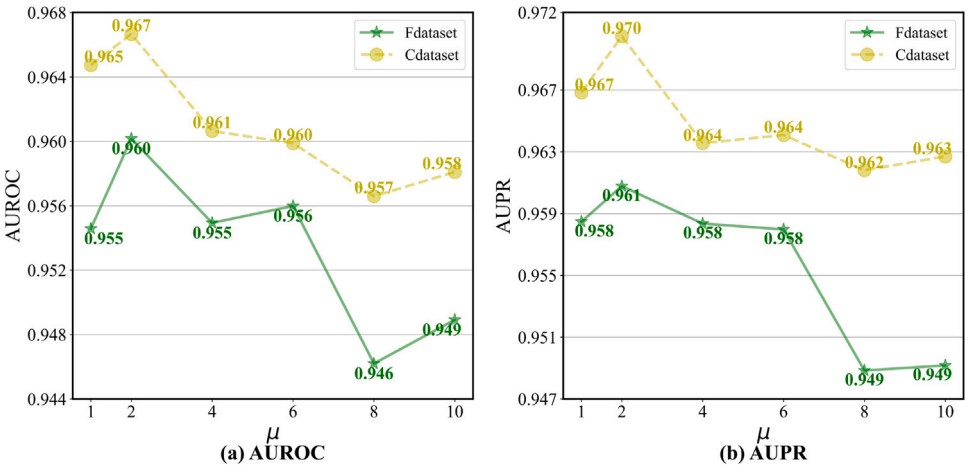

Fig. 5. Parameter sensitivity of the penalty factor μ on the Fdataset and Cdataset.

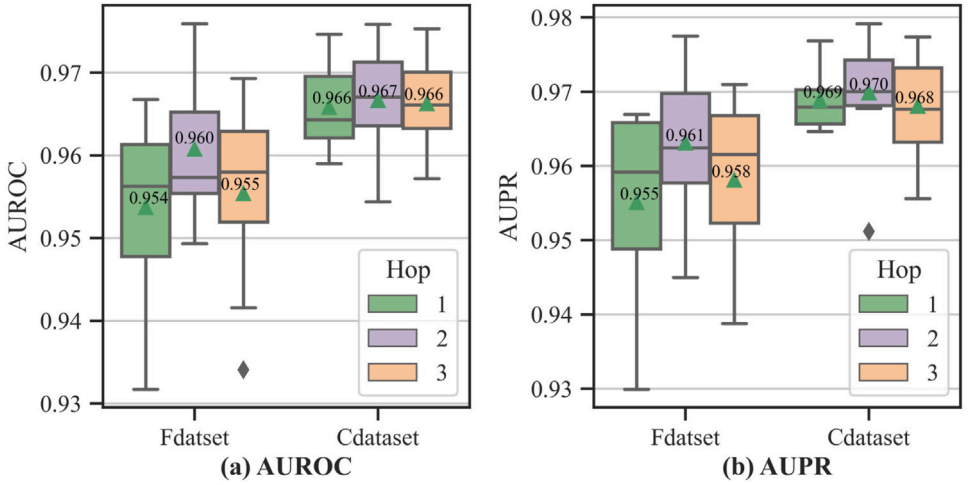

Fig. 6. Box plots of AUROC (a) and AUPR (b) under different hops. Green triangles denotes the mean values.

MPNNs on multiple networks to reduce the network-specific information loss, causing better performance than other models based on GNNs. Although SLGCN is based on the line graphs extracted from the heterogeneous network, it still shows a great improvement. This indicates that SLGCN can mine more meaningful messages from heterogeneous information.

PSGCN, LAGCN and DRWCNCF are also based on GCN. Their close results show that they are equally matched in predicting unknown associations. Although they perform better than SCPMF and MBiRW, their AUROC scores and AUPR scores are all lower than those of DRRS. These results show that SLGCN aggregates more useful neighboring messages and captures more comprehensive topology information around predicted associations than other methods based on GCN. Likewise, SLGCN shows its superiority over other methods on the Cdataset in Figs. 7 and 8.

It is a remarkable fact that the AUROC scores are less influenced by the ratio of positive samples to negative samples. From the definition of AUPR, AUPR scores decrease when the number of negative samples increases. A previous study [47] demonstrated that AUPR provides more information for identifying positive samples. From Table 2, the AUPR scores of SLGCN are highest on the two datasets. In addition, the AUPR scores for the unbalanced ratio are only 1.5% (on the Fdataset) and 1.7% (on the Cdataset) lower than those for the balanced ratio. The decline of SLGCN is less than those of the other models, proving that SLGCN can more precisely identify positive samples and is more robust when data change.

3.5. Ablation analysis

3.5.1. Analysis of graph transformation

In our work, there are two steps for constructing line graphs. First, enclosing subgraphs around predicted associations are extracted from the heterogeneous network (HN). Second, line graphs (LG) are transformed from enclosing subgraphs (ES). In practice, GNNs can also be applied on the enclosing subgraph (e.g., PSGCN) and the heterogeneous network (e.g., LAGCN). To demonstrate the efficiency of constructing line graphs, GCN [21], GAT [34], and GraphSAGE [32] are performed on three graphs. To ensure a fair comparison, three models have the same architecture with three layers and 128 units. For classification, the node features are fed to a fully connected layer. It is worth noting that when using GNNs on a heterogeneous network, a combination method is needed to integrate both drug and disease features before classification. The Hadamard product is utilized as the combination method. Additionally, when using GNNs on subgraphs, a pooling operation is necessary to obtain subgraph-level representations for classification. Max pooling is employed for this purpose. On line graphs, the target nodes' representation is directly used for prediction without additional operations.

Three types of GNNs are briefly presented as follows:

- GCN (graph convolutional network) [21]: uses an average neighborhood aggregation strategy to aggregate information from a node's neighbors and update its representation.

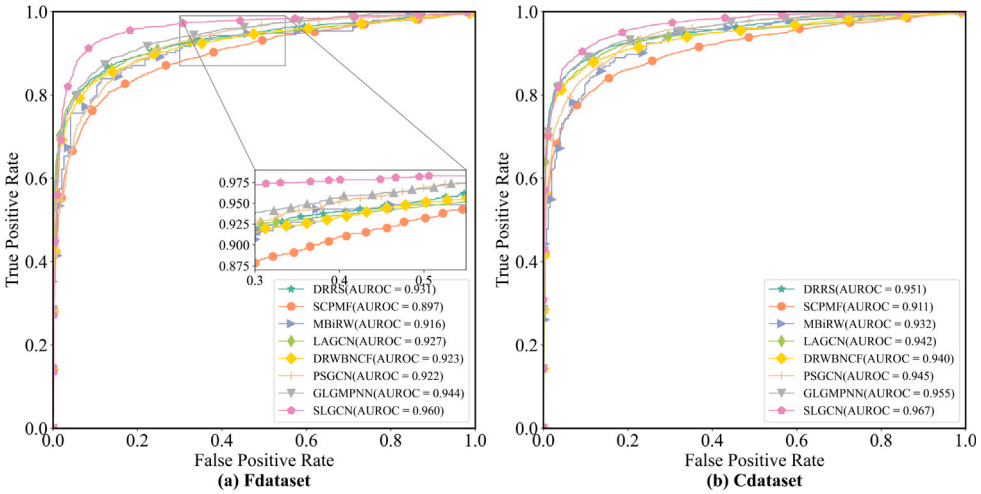

Fig. 7. ROC curves of all models on the Fdataset (a) and Cdataset (b) (neg:pos=1:1).

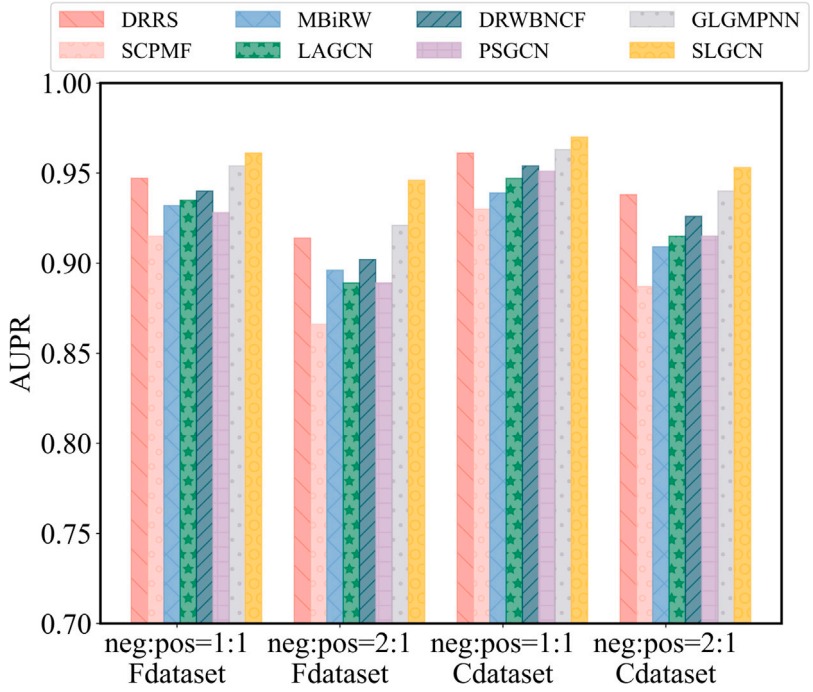

Fig. 8. AUPR scores obtained by all models under different sample ratios on the Fdataset and Cdataset.

Table 2
The AUROCs and AUPRs of all models in 10-CV.

Model	Fdataset				Cdataset			
	1:1		2:1		1:1		2:1	
	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR
DRRS	0.931 ± 0.013	0.947 ± 0.008	0.931 ± 0.015	0.914 ± 0.008	0.951 ± 0.009	0.961 ± 0.005	0.951 ± 0.009	0.938 ± 0.008
SCPMF	0.897 ± 0.015	0.915 ± 0.016	0.897 ± 0.012	0.866 ± 0.011	0.911 ± 0.013	0.930 ± 0.012	0.910 ± 0.012	0.887 ± 0.014
MBiRW	0.916 ± 0.013	0.932 ± 0.010	0.915 ± 0.015	0.896 ± 0.020	0.932 ± 0.015	0.939 ± 0.013	0.931 ± 0.009	0.909 ± 0.011
LAGCN	0.927 ± 0.013	0.935 ± 0.012	0.922 ± 0.014	0.889 ± 0.014	0.942 ± 0.007	0.947 ± 0.008	0.941 ± 0.006	0.915 ± 0.011
DRWBNCF	0.923 ± 0.009	0.940 ± 0.007	0.921 ± 0.012	0.902 ± 0.148	0.940 ± 0.012	0.954 ± 0.011	0.940 ± 0.010	0.926 ± 0.009
PSGCN	0.922 ± 0.009	0.928 ± 0.006	0.928 ± 0.013	0.889 ± 0.016	0.945 ± 0.006	0.951 ± 0.005	0.944 ± 0.006	0.915 ± 0.006
GLGMPNN	0.944 ± 0.011	0.954 ± 0.008	0.942 ± 0.010	0.921 ± 0.016	0.955 ± 0.008	0.963 ± 0.006	0.956 ± 0.007	0.940 ± 0.009
SLGCN	**0.960 ± 0.008**	**0.961 ± 0.008**	**0.962 ± 0.009**	**0.946 ± 0.015**	**0.967 ± 0.006**	**0.970 ± 0.007**	**0.969 ± 0.007**	**0.953 ± 0.010**

Note: The best results are in bold.

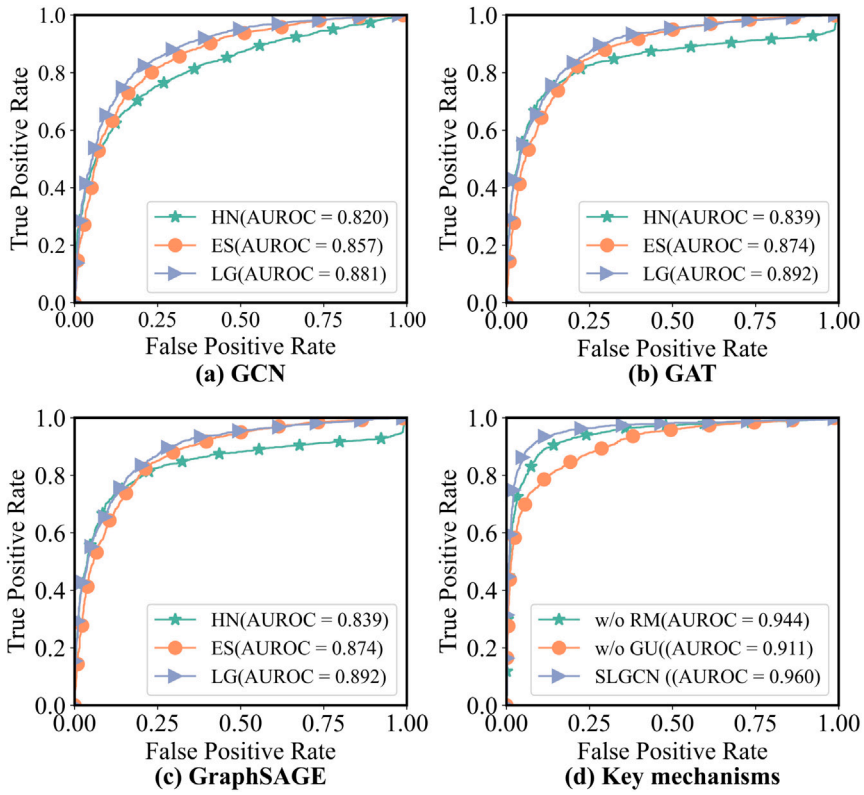

Fig. 9. (a–c) ROC curves obtained by GCN, GAT, GraphSAGE on three types of graphs; (d) ROC curves obtained by SLGCN and two variants on the Fdataset (neg:pos=1:1).

Table 3
Analysis of graph transformation to GNNs on the Fdataset (neg:pos=1:1).

Metrics	Models	HN	ES	LG
AUROC	GCN	0.820 ± 0.030	0.857 ± 0.015	**0.881 ± 0.015**
	GAT	0.839 ± 0.025	0.874 ± 0.011	**0.892 ± 0.015**
	GraphSAGE	0.855 ± 0.020	0.882 ± 0.015	**0.903 ± 0.010**
AUPR	GCN	0.838 ± 0.024	0.842 ± 0.027	**0.869 ± 0.020**
	GAT	0.869 ± 0.021	0.864 ± 0.018	**0.894 ± 0.015**
	GraphSAGE	0.856 ± 0.019	0.874 ± 0.023	**0.905 ± 0.012**

Note: The best results are in bold.

- GAT (graph attention network) [34]: uses self-attention mechanisms to assign different attention weights to different neighbors of a node, allowing it to focus on the most relevant neighbors while aggregating information.
- GraphSAGE (graph sample and aggregate) [32]: operates in a multilayer pattern, where each layer samples and aggregates information from a node's neighbors. The sampled information is then combined and used to update the nodes representation.

As observed in Table 3, employing GNNs on ES yields superior performance compared to using GNNs on HN. This result strongly suggests that subgraph extraction effectively enhances the feature learning capability of GNNs. Subgraph extraction prevents information oversquashing, a phenomenon that GNNs suffer, and reduces the risk of contaminating the representations with irrelevant information in deeper GNNs [48]. Furthermore, GNNs applied on LG achieve superior performance compared to the other two graphs, indicating that transformation to line graphs improves the ability of GNNs even further. The consistent improvement can be attributed to two key factors. First, the absence of pooling operations helps to minimize semantic information loss. Second, the preserved connectivity information in line graphs is valuable for capturing important patterns and features, thereby enhancing the efficiency of predicting drug–disease associations.

Table 4
Ablation study of key mechanisms of SLGCN on the Fdataset (neg:pos=1:1).

Metrics	SLGCN	SLGCN (w/o RM)	SLGCN (w/o GU)
AUROC	**0.960 ± 0.008**	0.944 ± 0.009	0.911 ± 0.008
AUPR	**0.961 ± 0.008**	0.949 ± 0.008	0.920 ± 0.011

Note: The best results are in bold.

3.5.2. Analysis of key mechanisms in SLGCN

To assess the effectiveness of the relation matrix (RM) and the gated update function (GU) in SLCGN, two variants are constructed. The brief introduction is as follows:

- SLGCN (w/o RM): SLGCN without the relation matrix. In message aggregation, no additional structural importance information is introduced.
- SLGCN (w/o GU): SLGCN without the gated update function. In the update function, ego representations are directly mixed with passed messages.

The results of SLGCN and its two variants are shown in Table 4 and Fig. 9(d). It is apparent that SLGCN outperforms its variants on the Fdataset. Compared with SLGCN (w/o RM), SLGCN significantly improves the AUROC score by 1.6% and the AUPR score by 1.2%. These results demonstrate the substantial benefits of utilizing the relation matrix to identify more important nodes from a topological perspective, enabling comprehensive feature learning. The introduction of the relation matrix can be likened to the attention mechanism employed in GAT. However, unlike other attention mechanisms, the relation matrix leverages distance information to identify crucial and advantageous neighbors based on the graph structure rather than relying solely on node attributes. In addition, the superior performance of SLGCN over SLGCN (w/o GU) highlights that using the gate update function to integrate ego representations and aggregate neighbor representations plays a vital role in developing effective GNNs.

Table 5
Top 10 predicted drugs associated with Alzheimer's disease on the Fdataset.

Rank	Drug	Evidence
1	Methotrexate	CTD
2	Prednisone	CTD
3	Dexamethasone	CTD
4	Vincristine	CTD
5	Clonidine	CTD
6	Gabapentin	CTD
7	Propranolol	CTD
8	Triamcinolone	CTD
9	Naproxen	CTD/DB
10	Primidone	NA

Table 6
Top 10 predicted drugs associated with breast cancer on the Fdataset.

Rank	Drug	Evidence
1	Clonidine	CTD/DB
2	Prednisone	CTD/DB
3	Gabapentin	CTD/DB
4	Dexamethasone	CTD/DB/PubChem
5	Baclofen	CTD
6	Caffeine	CTD/DB/PubChem
7	Metoprolol	CTD
8	Bleomycin	CTD
9	Valproic acid	CTD/DB
10	Indomethacin	CTD/DB/PubChem

4. Case studies

To demonstrate the clinical meaning of our model, experiments on Alzheimer's disease (AD) and breast cancer were conducted to analyze the associated drugs. In this section, all known associations of the Fdataset are used for training. All unobserved associations of AD or breast cancer are predicted with scores. According to the predicted scores, the top 10 drugs are listed in Tables 5 and 6. Public datasets, including CTD [43], DB [39], and PubChem [49], are used to confirm the existence of the predicted associations.

AD is a brain disease resulting in amnestic cognitive impairment [50]. To date, the complex relationships between AD and genes are not well appreciated [51]. Due to the severe damage to short-term memory, metal ability and other functions of patients, finding effective drugs for AD is significant for humans [52,53]. Although many researchers have been investigating how to prevent beta-amyloid plaques (Aβ) and neurofibrillary tangles (NFTs), the two core pathologies of AD, they still have not found effective therapeutic methods. A recent study [54] showed that inflammation may play a primary role in exacerbating Aβ and NFT. In Table 5, methotrexate and dexamethasone are clinically used for the treatment of rheumatoid arthritis and inflammatory skin conditions, respectively. In a recent study [55], methotrexate showed a strong relationship in reducing the risk of dementia in a case-control study. In addition, it has been proven that the combination of dexamethasone and acyclovir can prevent Aβ oligomer-induced cognitive impairments and that dexamethasone might play a main role in attenuating the overexpression of proinflammatory cytokines (TNF-α and IL-6) in hippocampal regions [56]. Admittedly, there are no public studies clearly displaying the therapeutic abilities of primidone for AD, but it has shown the power to inhibit the kinase activity of RIPK1 (receptor-interacting serine/threonine protein kinase 1), a drug target for treating AD [57].

Breast cancer is a severe disease with a massively high incidence in women [58]. In the past decade, the incidence rate of breast cancer has increased 0.5% annually in the United States [59]. Despite the lower death rate in recent years, a variety of long-term sequelae are still great challenges for survivors, such as disfiguring skin ulceration after radiation treatment [60] and fertility loss after endocrine treatment [61]. Therefore, discovering novel drugs related to breast cancer is needed. Chemotherapy, which is a part of standard treatment in many western countries, has some side effects, such as vomiting and allergic reactions. To reduce these side effects, dexamethasone is often used before chemotherapy. In a study [62] that tested the clinical benefit of abiraterone acetate (AA) plus prednisone, prednisone was described to stabilize tumor growth or help maximize the effect of AA. Additionally, a previous study [63] investigated whether dexamethasone can inhibit MCF-7 (a breast cancer cell line) proliferation. Valproic acid can reduce the expression of miR-34a and miR-520 h, resulting in a decline in the survival of MCF-7 cells [64]. Moreover, an organotin indomethacin derivative has been shown to inhibit the expression of IL-6, a proinflammatory cytokine, which is beneficial in breast cancer patients [65].

5. Conclusion

In this paper, SLGCN is proposed to learn comprehensive representations of predicted associations. First, the enclosing subgraphs centered around predicted associations are extracted and then transformed to line graphs. Because of the inefficiency of adjacency information in line graphs, a relation matrix based on the distance information of the original subgraphs is introduced in message aggregation. The introduction of the relation matrix helps identify more important and beneficial neighbors from the perspective of topology. The experiments demonstrate that SLGCN has strong prediction ability and has good capacity to find potential drugs for diseases.

The limitation of SLGCN is its memory and time costs. Although the size of graphs has been reduced to a great extent, it remains challenging to extract and store the subgraphs, especially for large-scale graphs. On the one hand, there is an urgent need to develop more efficient methods for extracting and storing subgraphs. On the other hand, the introduction of the relation matrix processed by GraphSAGE is complex. In future work, it is important to investigate more efficient methods for simplifying the relation matrix while still retaining the essential structural information derived from the graph.

CRediT authorship contribution statement

Bao-Min Liu: Writing – original draft, Visualization, Software, Methodology, Data curation, Conceptualization. **Ying-Lian Gao:** Supervision, Formal analysis. **Feng Li:** Supervision, Formal analysis. **Chun-Hou Zheng:** Writing – review & editing, Supervision. **Jin-Xing Liu:** Writing – review & editing, Supervision, Funding acquisition.

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.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China [62172254].

References

[1] J.-P. Jourdan, R. Bureau, C. Rochais, P. Dallemagne, Drug repositioning: A brief overview, J. Pharm. Pharmacol. 72 (9) (2020) 1145–1151.

[2] N. Raje, K. Anderson, Thalidomide – a revival story, N. Engl. J. Med. 341 (21) (1999) 1606–1609.

[3] S. Pushpakom, F. Iorio, P.A. Eyers, K.J. Escott, S. Hopper, A. Wells, A. Doig, T. Guilliams, J. Latimer, C. McNamee, A. Norris, P. Sanseau, D. Cavalla, M. Pirmohamed, Drug repurposing: Progress, challenges and recommendations, Nat. Rev. Drug Discov. 18 (1) (2019) 41–58.

[4] J. Avorn, The $2.6 Billion Pill – Methodologic and policy considerations, N. Engl. J. Med. 372 (20) (2015) 1877–1879.

[5] S. Naylor, M. Kauppi, J.M. Schonfeld, Therapeutic drug repurposing, repositioning and rescue: Part II: Business review, Drug Discov. World 16 (2015) 57–72.

[6] Y. Ge, T. Tian, S. Huang, F. Wan, J. Li, S. Li, X. Wang, H. Yang, L. Hong, N. Wu, E. Yuan, Y. Luo, L. Cheng, C. Hu, Y. Lei, H. Shu, X. Feng, Z. Jiang, Y. Wu, Y. Chi, X. Guo, L. Cui, L. Xiao, Z. Li, C. Yang, Z. Miao, L. Chen, H. Li, H. Zeng, D. Zhao, F. Zhu, X. Shen, J. Zeng, An integrative drug repositioning framework discovered a potential therapeutic agent targeting COVID-19, Signal Transduct. Target. Ther. 6 (1) (2021) 165.

[7] F. Ahmed, A.M. Soomro, A.R.C. Salih, A. Samantasinghar, A. Asif, I.S. Kang, K.H. Choi, A comprehensive review of artificial intelligence and network based approaches to drug repurposing in Covid-19, Biomed. Pharmacother. 153 (2022) 113350.

[8] W. Zhang, H. Xu, X. Li, Q. Gao, L. Wang, DRIMC: An improved drug repositioning approach using Bayesian inductive matrix completion, Bioinformatics 36 (9) (2020) 2839–2847.

[9] J. Ha, SMAP: Similarity-based matrix factorization framework for inferring mirna-disease association, Knowl.-Based Syst. 263 (2023) 110295.

[10] W. Zhang, X. Yue, F. Huang, R. Liu, Y. Chen, C. Ruan, Predicting drug-disease associations and their therapeutic function based on the drug-disease association bipartite network, Methods 145 (2018) 51–59.

[11] X. Liang, P. Zhang, L. Yan, Y. Fu, F. Peng, L. Qu, M. Shao, Y. Chen, Z. Chen, LRSSL: Predict and interpret drug-disease associations based on data integration using sparse subspace learning, Bioinformatics 33 (8) (2017) 1187–1196.

[12] B. Wang, A.M. Mezlini, F. Demir, M. Fiume, Z. Tu, M. Brudno, B. Haibe-Kains, A. Goldenberg, Similarity network fusion for aggregating data types on a genomic scale, Nature Methods 11 (3) (2014) 333–337.

[13] P. Xuan, Y. Cao, T. Zhang, X. Wang, S. Pan, T. Shen, Drug repositioning through integration of prior knowledge and projections of drugs and diseases, Bioinformatics 35 (20) (2019) 4108–4119.

[14] W. Zhang, X. Yue, W. Lin, W. Wu, R. Liu, F. Huang, F. Liu, Predicting drug-disease associations by using similarity constrained matrix factorization, BMC Bioinform. 19 (1) (2018) 233.

[15] H. Luo, J. Wang, M. Li, J. Luo, X. Peng, F.X. Wu, Y. Pan, Drug repositioning based on comprehensive similarity measures and Bi-Random walk algorithm, Bioinformatics 32 (17) (2016) 2664–2671.

[16] M.M. Yin, J.X. Liu, Y.L. Gao, X.Z. Kong, C.H. Zheng, NCPLP: A novel approach for predicting microbe-associated diseases with network cnsistency pojection and label popagation, IEEE Trans. Cybern. 52 (6) (2022) 5079–5087.

[17] E. Noor, S. Cherkaoui, U. Sauer, Biological insights through omics data integration, Curr. Opin. Struct. Biol. 15 (2019) 39–47.

[18] L. Riva, S. Yuan, X. Yin, L. Martin-Sancho, N. Matsunaga, L. Pache, S. Burgstaller-Muehlbacher, P.D.D. Jesus, P. Teriete, M.V. Hull, M.W. Chang, J.F.-W. Chan, J. Cao, V.K.-M. Poon, K.M. Herbert, K. Cheng, T.-T.H. Nguyen, A. Rubanov, Y. Pu, C. Nguyen, A. Choi, R. Rathnasinghe, M. Schotsaert, L. Miorin, M. Dejosez, T.P. Zwaka, K.-Y. Sit, L. Martinez-Sobrido, W.-C. Liu, K.M. White, M.E. Chapman, E.K. Lendy, R.J. Glynne, R. Albrecht, E. Ruppin, A.D. Mesecar, J.R. Johnson, C. Benner, R. Sun, P.G. Schultz, A.I. Su, A. García-Sastre, A.K. Chatterjee, K.-Y. Yuen, S.K. Chanda, Discovery of SARS-CoV-2 antiviral drugs through large-scale compound repurposing, Nature 586 (7827) (2020) 113–119.

[19] C. Outeiral, M. Strahm, J. Shi, G.M. Morris, S.C. Benjamin, C.M. Deane, The prospects of quantum computing in computational molecular biology, Wiley Interdiscip. Rev. Comput. Mol. Sci. 11 (1) (2021) e1481.

[20] C. Angermueller, T. Pärnamaa, L. Parts, O. Stegle, Deep learning for computational biology, Mol. Syst. Biol. 12 (7) (2016) 878.

[21] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, in: Int. Conf. Learn. Represent., 2017, pp. 1–14.

[22] H. Fu, F. Huang, X. Liu, Y. Qiu, W. Zhang, MVGCN: Data integration through multi-view graph convolutional network for predicting links in biomedical bipartite networks, Bioinformatics 38 (2) (2021) 426–434.

[23] H. Yang, Y. Ding, J. Tang, F. Guo, Inferring human microbe–drug associations via multiple kernel fusion on graph neural network, Knowl.-Based Syst. 238 (2022) 107888.

[24] Z. Yu, F. Huang, X. Zhao, W. Xiao, W. Zhang, Predicting drug–disease associations through layer attention graph convolutional network, Brief. Bioinform. 22 (4) (2021) bbaa243.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Kaiser, I. Polosukhin, Attention is all you need, in: Proc. 31st Int. Conf. Neural Inf. Process. Syst., 2017, pp. 6000–6010.

[26] Y. Meng, C. Lu, M. Jin, J. Xu, X. Zeng, J. Yang, A weighted bilinear neural collaborative filtering approach for drug repositioning, Brief. Bioinform. 23 (2) (2022) bbab581.

[27] M. Zhang, Y. Chen, Link prediction based on graph neural networks, in: Proc. 32nd Int. Conf. Neural Inf. Process. Syst., 2018, pp. 5171–5181.

[28] X. Sun, B. Wang, J. Zhang, M. Li, Partner-specific drug repositioning approach based on graph convolutional network, IEEE J. Biomed. Health Inform. 26 (11) (2022) 5757–5765.

[29] L. Cai, J. Li, J. Wang, S. Ji, Line graph neural networks for link prediction, IEEE Trans. Pattern Anal. Mach. Intell. 44 (9) (2022) 5103–5113.

[30] H. Fan, F. Zhang, Y. Wei, Z. Li, C. Zou, Y. Gao, Q. Dai, Heterogeneous hypergraph variational autoencoder for link prediction, IEEE Trans. Pattern Anal. Mach. Intell. 44 (8) (2022) 4125–4138.

[31] Z. Wang, Y. Chai, C. Sun, X. Rui, H. Mi, X. Zhang, P.S. Yu, A weighted symmetric graph embedding approach for link prediction in undirected graphs, IEEE Trans. Cybern. early access, http://dx.doi.org/10.1109/TCYB.2022.3181810.

[32] W.L. Hamilton, R. Ying, J. Leskovec, Inductive representation learning on large graphs, in: Proc. 31st Int. Conf. Neural Inf. Process. Syst., 2017, pp. 1025–1035.

[33] C. Gao, J. Zhu, F. Zhang, Z. Wang, X. Li, A novel representation learning for dynamic graphs based on graph convolutional networks, IEEE Trans. Cybern. early access, http://dx.doi.org/10.1109/TCYB.2022.3159661.

[34] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio', Y. Bengio, Graph attention networks, in: Proc. 6th Int. Conf. Learn. Represent, 2018, pp. 1–12.

[35] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: Int. Conf. Learn. Represent., 2015, pp. 1–14.

[36] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929–1958.

[37] A. Gottlieb, G.Y. Stein, E. Ruppin, R. Sharan, PREDICT: A method for inferring novel drug indications with application to personalized medicine, Mol. Syst. Biol. 7 (1) (2011) 496.

[38] D. Weininger, SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules, J. Chem. Inf. Comput. Sci. 28 (1) (1988) 31–36.

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

[40] C. Steinbeck, C. Hoppe, S. Kuhn, M. Floris, R. Guha, E.L. Willighagen, Recent developments of the chemistry development kit (CDK) - an open-source java library for chemo- and bioinformatics, Curr. Pharm. Des. 12 (17) (2006) 2111–2120.

[41] M.A. van Driel, J. Bruggeman, G. Vriend, H.G. Brunner, J.A. Leunissen, A text-mining analysis of the human phenome, Eur. J. Hum. Genet. 14 (5) (2006) 535–542.

[42] A. Hamosh, A.F. Scott, J.S. Amberger, C.A. Bocchini, V.A. McKusick, Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders, Nucleic Acids Res. 33 (Database issue) (2005) D514–D517.

[43] A.P. Davis, C.J. Grondin, R.J. Johnson, D. Sciaky, J. Wiegers, T.C. Wiegers, C.J. Mattingly, Comparative toxicogenomics database (CTD): Update 2021, Nucleic Acids Res. 49 (D1) (2021) D1138–D1143.

[44] H. Luo, M. Li, S. Wang, Q. Liu, Y. Li, J. Wang, Computational drug repositioning using low-rank matrix approximation and randomized algorithms, Bioinformatics 34 (11) (2018) 1904–1912.

[45] Y. Meng, M. Jin, X. Tang, J. Xu, Drug repositioning based on similarity constrained probabilistic matrix factorization: COVID-19 as a case study, Appl. Soft Comput. 103 (2021) 107135.

[46] B.-M. Liu, Y.-L. Gao, D.-J. Zhang, F. Zhou, J. Wang, C.-H. Zheng, J.-X. Liu, A new framework for drug–disease association prediction combing light-gated message passing neural network and gated fusion mechanism, Brief. Bioinform. (2022) bbac457.

[47] K. Pliakos, C. Vens, Network inference with ensembles of bi-clustering trees, BMC Bioinform. 20 (1) (2019) 525.

[48] H. Yin, M. Zhang, Y. Wang, J. Wang, P. Li, Algorithm and system co-design for efficient subgraph-based graph representation learning, in: Proc. VLDB Endow., Vol. 15, 2022, pp. 2788–2796.

[49] S. Kim, J. Chen, T. Cheng, A. Gindulyte, J. He, S. He, Q. Li, B.A. Shoemaker, P.A. Thiessen, B. Yu, PubChem 2019 update: Improved access to chemical data, Nucleic Acids Res. 47 (D1) (2019) D1102–D1109.

[50] D.S. Knopman, H. Amieva, R.C. Petersen, G. Chételat, D.M. Holtzman, B.T. Hyman, R.A. Nixon, D.T. Jones, Alzheimer disease, Nat. Rev. Dis. Primers 7 (1) (2021) 33.

[51] Y.H. Kim, S.H. Beak, A. Charidimou, M. Song, Discovering new genes in the pathways of common sporadic neurodegenerative diseases: A bioinformatics approach, J. Alzheimers Dis. 51 (1) (2016) 293–312.

[52] L.-K. Huang, S.-P. Chao, C.-J. Hu, Clinical trials of new drugs for Alzheimer disease, J. Biomed. Sci. 27 (1) (2020) 18.

[53] A. Judge, C. Garriga, N.K. Arden, S. Lovestone, D. Prieto-Alhambra, C. Cooper, C.J. Edwards, Protective effect of antirheumatic drugs on dementia in rheumatoid arthritis patients, Alzheimers Dement (N Y) 3 (4) (2017) 612–621.

[54] J.W. Kinney, S.M. Bemiller, A.S. Murtishaw, A.M. Leisgang, A.M. Salazar, B.T. Lamb, Inflammation as a central mechanism in Alzheimer's disease, Alzheimers Dement (N Y) 4 (1) (2018) 575–590.

[55] D. Newby, D. Prieto-Alhambra, T. Duarte-Salles, D. Ansell, L. Pedersen, J. van der Lei, M. Mosseveld, P. Rijnbeek, G. James, M. Alexander, P. Egger, J. Podhorna, R. Stewart, G. Perera, P. Avillach, S. Grosdidier, S. Lovestone, A.J. Nevado-Holgado, Methotrexate and relative risk of dementia amongst patients with rheumatoid arthritis: a multi-national multi-database case-control study, Alzheimers Res. Ther. 12 (1) (2020) 38.

[56] H. Zhang, Z. Yuan, Y. Yan, L. Chen, Y. Zhou, D. Zhang, C.C. Tony, W. Cui, The combination of acyclovir and dexamethasone protects against Alzheimer's disease-related cognitive impairments in mice, Psychopharmacology 237 (6) (2020) 1851–1860.

[57] T. Riebeling, K. Jamal, R. Wilson, B. Kolbrink, F.A. von Samson-Himmelstjerna, C. Moerke, L.R. Garcia, E. Dahlke, F. Michels, F. Lühder, D. Schunk, P. Doldi, B. Tyczynski, A. Kribben, C. Flüh, F. Theilig, U. Kunzendorf, P. Meier, S. Krautwald, Primidone blocks RIPK1-driven cell death and inflammation, Cell Death Differ. 28 (5) (2021) 1610–1626.

[58] A.G. Waks, E.P. Winer, Breast cancer treatment: A review, JAMA 321 (3) (2019) 288–300.

[59] A.N. Giaquinto, H. Sung, K.D. Miller, J.L. Kramer, L.A. Newman, A. Minihan, A. Jemal, R.L. Siegel, Breast cancer statistics, 2022, CA Cancer J. Clin. 72 (6) (2022) 524–541.

[60] E.C. Milam, L.K. Rangel, M.K. Pomeranz, Dermatologic sequelae of breast cancer: From disease, surgery, and radiation, Int. J. Dermatol. 60 (4) (2021) 394–406.

[61] D.L. Lovelace, L.R. McDaniel, D. Golden, Long-term effects of breast cancer surgery, treatment, and survivor care, J. Midwifery Womens Health 64 (6) (2019) 713–724.

[62] H. Bonnefoi, T. Grellety, O. Tredan, M. Saghatchian, F. Dalenc, A. Mailliez, T. L'Haridon, P. Cottu, S. Abadie-Lacourtoisie, B. You, M. Mousseau, J. Dauba, F.D. Piano, I. Desmoulins, F. Coussy, N. Madranges, J. Grenier, F.C. Bidard, C. Proudhon, G. MacGrogan, C. Orsini, M. Pulido, A. Gonçalves, A phase II trial of abiraterone acetate plus prednisone in patients with triple-negative androgen receptor positive locally advanced or metastatic breast cancer (UCBG 12-1), Ann. Oncol. 27 (5) (2016) 812–818.

[63] F. Buxant, N. Kindt, G. Laurent, J.C. Nol, S. Saussez, Antiproliferative effect of dexamethasone in the MCF-7 breast cancer cell line, Mol. Med. Rep. 12 (3) (2015) 4051–4054.

[64] N. Injinari, Z. Amini-Farsani, M. Yadollahi-Farsani, H. Teimori, Apoptotic effects of valproic acid on mir-34a, mir-520h and HDAC1 gene in breast cancer, Life Sci. 269 (2021) 119027.

[65] M. Segovia-Mendoza, C. Camacho-Camacho, I. Rojas-Oviedo, H. Prado-Garcia, D. Barrera, I. Martínez-Reza, F. Larrea, R. García-Becerra, An organotin indomethacin derivative inhibits cancer cell proliferation and synergizes the antiproliferative effects of lapatinib in breast cancer cells, Am. J. Cancer Res. 10 (10) (2020) 3358–3369.