This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2021.3059415, IEEE/ACM Transactions on Computational Biology and Bioinformatics

Predicting Biomedical Interactions with Higher-Order Graph Convolutional Networks

Kishan KC, Rui Li, Feng Cui, and Anne R. Haake

Abstract—Biomedical interaction networks have incredible potential to be useful in the prediction of biologically meaningful interactions, identification of network biomarkers of disease, and the discovery of putative drug targets. Recently, graph neural networks have been proposed to effectively learn representations for biomedical entities and achieved state-of-the-art results in biomedical interaction prediction. These methods only consider information from immediate neighbors but cannot learn a general mixing of features from neighbors at various distances. In this paper, we present a higher-order graph convolutional network (HOGCN) to aggregate information from the higher-order neighborhood for biomedical interaction prediction. Specifically, HOGCN collects feature representations of neighbors at various distances and learns their linear mixing to obtain informative representations of biomedical entities. Experiments on four interaction networks, including protein-protein, drug-drug, drug-target, and gene-disease interactions, show that HOGCN achieves more accurate and calibrated predictions. HOGCN performs well on noisy, sparse interaction networks when feature representations of neighbors at various distances are considered. Moreover, a set of novel interaction predictions are validated by literature-based case studies.

Index Terms—Higher-Order Graph Convolution, Interaction Prediction, Biomedical interaction networks.

1 Introduction

A Biological system is a complex network of various molecular entities such as genes, proteins, and other biological molecules linked together by the interactions between these entities. The complex interplay between various molecular entities can be represented as interaction networks with molecular entities as nodes and their interactions as edges. Such a representation of a biological system as a network provides a conceptual and intuitive framework to investigate and understand direct or indirect interactions between different molecular entities in a biological system. Study of such networks lead to system-level understanding of biology [1] and discovery of novel interactions including protein-protein interactions (PPIs) [2], drug-drug interactions (DDIs) [3], drug-target interactions (DTIs) [4] and gene-disease associations (GDIs) [5].

Recently, the generalization of deep learning to the network-structured data [6] has shown great promise across various domains such as social networks [7], recommendation systems [8], chemistry [9], citation networks [10]. These approaches are under the umbrella of graph convolutional networks (GCNs). GCNs repeatedly aggregate feature representations of immediate neighbors to learn the informative representation of the nodes for link prediction. Although GCN based methods show great success in biomedical interaction prediction [3], [11], the issue with such methods is that they only consider information from immediate neighbors. SkipGNN [12] leverages a skip graph to aggregate feature representations from direct and second-order neighbors and demonstrated improvements over GCN methods in biomedical interaction prediction. However, SkipGNN cannot be applied to aggregate information from higher-order neighbors and thus fail to capture information that resides farther away from a particular interaction [13].

To address the challenge, we propose an end-to-end deep graph representation learning framework named higher-order graph convolutional networks (HOGCN) for predicting interactions between pairs of biomedical entities. HOGCN learns a representation for every biomedical entity using an interaction network structure $\mathcal{G}$ and/or features $\mathbf{X}$. In particular, we adopt a higher-order graph convolution (HOGC) layer [13] where the feature representations from k-order neighbors are considered to obtain the representation of biomedical entities. The layer can thus learn to mix feature representations of neighbors at various distances for interaction prediction. Furthermore, we define a bilinear decoder to reconstruct the edges in the input interaction network $\mathcal{G}$ by relying on feature representations produced by HOGC layers. The encoder-decoder approach makes HOGCN an end-to-end trainable model for interaction prediction.

We compare HOGCN's performance with that of state-of-the-art network similarity-based methods [14], network embedding methods [15], [16], and graph convolution-based methods [10], [12], [17] for biomedical link prediction. We experiment with various interaction datasets and show that our method makes accurate and calibrated predictions. HOGCN outperforms alternative methods based on network embedding by up to 30%. Furthermore, HOGCN outperforms graph convolution-based methods by up to 6%, alluding to the benefits of aggregating information from higher-order neighbors.

We perform a case study on the DDI network and

- *K. KC, R. Li, and A. Haake are with the Department of Computing and Information Sciences, Rochester Institute of Technology, Rochester, NY, 14623.*
  E-mail: {kk3671, arhics, rxlics}@rit.edu
- *F. Cui is with Thomas H. Gosnell School of Life Sciences, Rochester Institute of Technology, Rochester, NY, 14623. E-mail: fxcsbi@rit.edu*

observe that aggregating information from a higher-order neighborhood allows HOGCN to learn meaningful representation for drugs. Moreover, literature-based case studies illustrate that the novel predictions are supported by evidence, suggesting that HOGCN can identify interactions that are highly likely to be true positive.

In summary, our study demonstrates the ability of HOGCN to identify potential interactions between biomedical entities and opens up the opportunities to use the biological and physicochemical properties of biomedical entities for follow-up analysis of these interactions. Our code is available on https://github.com/kckishan/HOGCN-LP.

2 Related works

With the increasing coverage of the interactome, various network-based approaches have been proposed to exploit already available interactions to predict missing interactions [14], [18]–[20]. These methods can be broadly classified into (1) network similarity-based methods (2) network embedding methods (3) graph convolution-based methods. We next summarize these categories of methods for biomedical interaction prediction.

Given a network of known interactions, various similarity metrics are used to measure the similarity between the biomedical entities [18] with an assumption that higher similarity indicates interaction. Triadic closure principle (TCP) has been explored in biomedical interaction prediction with the hypothesis that biomedical entities with common interaction partners are likely to interact with each other [19]. TCP relies on a common neighbor algorithm to count the number of shared neighbors between the nodes and is quantified by $\mathbf{A}^2$ where $\mathbf{A}$ is the adjacency matrix. Recently, L3 heuristic [14] shows the common neighbor hypothesis fails for most protein pairs in PPI prediction and proposes to consider nodes that are similar to the neighbors of the nodes and can be quantified by $\mathbf{A}^3$. This indicates that higher-order neighbors are important for interaction prediction.

Next, network embedding methods embed the existing networks to low-dimensional space that preserves the structural proximity such that the nodes in the original network can be represented as low-dimensional vectors. Deepwalk [15] is a popular approach that generates the truncated random walks in the network and defines a neighborhood for each node as a set of nodes within a window of size k in each random walk. Similarly, node2vec performs a biased random walk by balancing the breadth-first and depth-first search in the network. The random walks generated by these methods can be considered as a combination of nodes from various order of neighborhoods such as 1-hop to k-hop neighborhood. In other words, DeepWalk and node2vec learn the embeddings for the nodes in the network from the combination of $\mathbf{A}^1, \mathbf{A}^2, \mathbf{A}^3, \ldots \mathbf{A}^k$ where $\mathbf{A}^i$ is the i^{th} power of the adjacency matrix. These embeddings can then be fed into a classifier to predict the interaction probability. These methods are only limited to the structure of the biomedical networks and cannot incorporate additional information about the biomedical entities. Also, they cannot learn the feature difference between nodes at various distances.

Furthermore, graph convolution-based methods use a message-passing mechanism to receive and aggregate information from neighbors to generate representations for the nodes in the network. Graph convolutional networks (GCNs) [17] and variational graph convolutional autoencoder (VGAE) [10] aggregate feature representation from immediate neighbors to learn the representation of biomedical entities in an end-to-end manner using link prediction objective. These methods are only limited to the average pooling of the neighborhood features [13]. SkipGNN [12] therefore proposes to use skip similarity between the biomedical entities to aggregate information from second-order neighbors. Moreover, graph attention network (GAT) proposed the convolution operation based on weighted attention mechanism to learn different weights for the neighbors in the first-order neighborhood. Since the convolution operation is dependent on node features $\mathbf{X}$, the lack of node features limits the applicability of GAT in interaction prediction task using network structure information only. Furthermore, a thorough empirical evaluation of GCN and GAT in [21] demonstrated that GCN performs better than GAT if hyperparameter and the training procedure are tuned fairly for all models. Thus, we have not been considered GAT for comparison.

Since the effect on a biological entity can be propagated to influence its neighbors at different distances in different ways and thus can determine system-level characteristics of the biological system, it is important to capture such influences at different distances. Although GCN-based methods can stack multiple GCN layers to aggregate information from neighbors at various distances, MixHop [13] theoretically demonstrated that GCN-based methods cannot learn linear combination of features (that includes subtraction *i.e.* $f(\sigma(\widehat{\mathbf{A}}\mathbf{X}) - \sigma(\widehat{\mathbf{A}}^2\mathbf{X})))$ between neighbors at various distances. Thus, we adopt a higher-order graph convolution (HOGC) layer from MixHop [13] to learn such feature differences and encode the higher-order network properties of biomedical networks. Next, we propose a novel bilinear decoder to model the interactions between biological entities using the feature representation learned by the encoder with the HOGC layer.

3 Preliminaries

A biomedical network is defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ where $\mathcal{V}$ denotes the set of nodes representing biomedical entities (e.g. proteins, genes, drugs, diseases) and $|\mathcal{V}|$ denotes the number of nodes. $\mathcal{E} \subseteq (\mathcal{V} \times \mathcal{V})$ denotes a set of interactions between biomedical entities. $\mathbf{X} \in \mathbb{R}^{|\mathcal{V}| \times F}$ is the features of biomedical entities where F is the dimension of features.

Let $\mathbf{A}$ denote the adjacency matrix of $\mathcal{G}$, where A_{ij} indicates an edge between nodes v_i and v_j. We assume the case of binary adjacency matrix $A_{ij} \in \{0, 1\}^{n \times n}$ where A_{ij} represents the existence of edge between the nodes v_i and v_j, indicating the presence of the experimental evidence for their interaction (i.e. $A_{ij} = 1$) or the absence of the experimental evidence for their interaction (i.e. $A_{ij} = 0$). Note that the same notation of adjacency matrix can be used to represent weighted graphs such that $A_{ij} = [0, 1]$. Table 1 shows the notations and their definitions used in the paper.

TABLE 1
Terms and notations

Notation	Definition
$\mathcal{G} : \{\mathcal{V}, \mathcal{E}, \mathbf{X}\}$	Graph with nodes $\mathcal{V}$, edges $\mathcal{E}$ and features $\mathbf{X}$
$\mathcal{E}'$	Test edges
$\mathbf{A} \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{V}\rvert}$	Adjacency matrix of graph $\mathcal{G}$
$\mathbf{D} \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{V}\rvert}$	Degree matrix with $D_{ii} = \sum_i A_{ij}$
$\mathbf{I} \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{V}\rvert}$	Identity matrix
$\widehat{\mathbf{A}} \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times \lvert\mathcal{V}\rvert}$	Symmetrically normalized adjacency matrix
$\mathbf{A} \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times F}$	F-dimensional feature matrix
$A_{ij} \in \{0, 1\}$	Ground-truth interaction between nodes i and j
$p_{ij} \in [0, 1]$	Probability of interaction between nodes i and j
$\mathbf{Z} \in \mathbb{R}^{\lvert\mathcal{V}\rvert \times d^*}$	Final node embeddings
$\mathbf{W}_{(l)}^{(i)}$	Weight matrix for i^{th} adjacency power for layer l
L	Number of HOGC layers
T	Number of training epochs
k	The order of neighborhood
$\mathbb{P}$	A set of integer adjacency powers $\mathbb{P} = \{0, 1, \ldots, k\}$
$\mathbf{O}_j^{(l)}$	Representation of neighbors at distance j in layer l

Problem Statement. (Biomedical interaction prediction) Given a biomedical interaction network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ and the set of potential biomedical interactions $\mathcal{E}'$, we aim to learn a interaction prediction model f to predict the interaction probabilities of $\mathcal{E}'$, $f : \mathcal{E}' \to [0, 1]$.

3.1 Message Passing

Given a biomedical network $\mathcal{G}$, message passing algorithms learn the representation of biomedical entities in the network by aggregating information from immediate neighbors [9]. Additional information about biomedical entities can be used to initialize the feature matrix $\mathbf{X}$. These algorithms involve the message passing step in which each biomedical entity sends its current representation to, and aggregates incoming messages from its immediate neighbors. Representation for each biomedical entity can be obtained after L steps of message passing and feature aggregation. However, such message passing operation is limited to average pooling of features from immediate neighbors and thus is unable to learn feature differences among neighbors at various distances [13].

Neighborhood nodes at various distances provide network structure information at different granularities [22]–[26]. Taking k-hop neighborhoods into consideration, we aim at aggregating information from various distances at every message passing step. Different powers of adjacency matrices such as $\mathbf{A}^1, \mathbf{A}^2, \mathbf{A}^3, \ldots, \mathbf{A}^k$ provide information about the network structure at different scales. Higher-order message passing operations can therefore learn to mix their representations using various powers of the adjacency matrix at each message passing step.

3.2 Graph Convolutional Networks (GCNs)

Graph convolutional networks (GCNs) are the generalization of convolution operation from regular grids such as images or texts to graph structured data [6], [27]. The key idea of GCNs is to learn the function to generate the node's representation by repeatedly aggregating information from immediate neighbors. The graph convolutional layer is defined as:

$$\mathbf{H}^{(l)} = \sigma(\widehat{\mathbf{A}}\mathbf{H}^{(l-1)}\mathbf{W}^{(l)}) \tag{1}$$

where $\mathbf{H}^{(l-1)}$ and $\mathbf{H}^{(l)}$ are the input and output activations, $\mathbf{W}^{(l)}$ is a trainable weight matrix of the layer l, σ is the element-wise activation, and $\widehat{\mathbf{A}}$ is a symmetrically normalized adjacency matrix with self-connections $\widehat{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{A} + \mathbf{I}_{|\mathcal{V}|})\mathbf{D}^{-\frac{1}{2}}$. A GCN model with L layers is then defined as:

$$\mathbf{H}^{(l)} = \begin{cases} \mathbf{X} & \text{if } l = 0 \\ \sigma(\widehat{\mathbf{A}}\mathbf{H}^{(l-1)}\mathbf{W}^{(l)}) & \text{if } l \in [1, \ldots, L] \end{cases}$$

and $\mathbf{H}^{(L)}$ can be used to predict the probability of interactions between biomedical entities.

4 Higher-order Graph Convolution Network (HOGCN)

In this work, we develop a higher-order graph convolutional network (HOGCN) that takes an interaction network $\mathcal{G}$ as input and reconstruct the edges in the interaction network (Fig. 1). HOGCN has two main components:

- **Encoder**: a higher-order graph convolution encoder that operates on an interaction graph $\mathcal{G}$ and produces representations for biomedical entities by aggregating features from the neighborhood at various distances and
- **Decoder**: a bilinear decoder that relies on these representations to reconstruct the interactions in $\mathcal{G}$.

4.1 Higher-Order Graph Encoder

We first describe the higher-order graph encoder, that operates on an undirected interaction graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ and learns the representations for biomedical entities.

We develop a higher-order graph encoder to capture higher-order features of the biological network. In particular, we adopt higher-order graph convolution layer (HOGC) [13] as a component of the encoder to capture such features and learn linear mixing of feature representations from neighbors at k-distances. HOGC layer is defined as:

$$\mathbf{H}^{(l)} = \underset{j \in \mathbb{P}}{\Big\|} \sigma\left(\widehat{\mathbf{A}}^j \mathbf{H}^{(l-1)} \mathbf{W}_j^{(l)}\right) \tag{2}$$

where $\mathbb{P}$ is a set of integer adjacency powers, $\widehat{\mathbf{A}}^j$ denotes the adjacency matrix $\widehat{\mathbf{A}}$ multiplied j times, and $\|$ denotes column-wise concatenation [13]. Graph convolutional network [17] only considers the 1st power of adjacency matrix and can be exactly recovered by setting $\mathbb{P} = \{1\}$ in Equation (2). Similarly, SkipGNN [12] considers direct and skip similarity and can be recovered by setting $\mathbb{P} = \{1, 2\}$ in Equation (2).

Fig. 2 shows a HOGC layer with $\mathbb{P} = \{0, 1, \ldots, k\}$ where k is maximum order of neighborhood considered in each HOGC layer. If $k = 0$, the HOGC layer only considers the features of the biomedical entities and can capture the feature similarity between various biological entities. This is equivalent to a fully connected network with features of biomedical entities as input. For the HOGC layer, $\widehat{\mathbf{A}}^0$ is the identity matrix $\mathbf{I}_{|\mathcal{V}|}$ where $|\mathcal{V}|$ is the number of nodes in the network. This allows the HOGC layer to learn the transformation of node features separately and mix it with feature representations from neighbors.

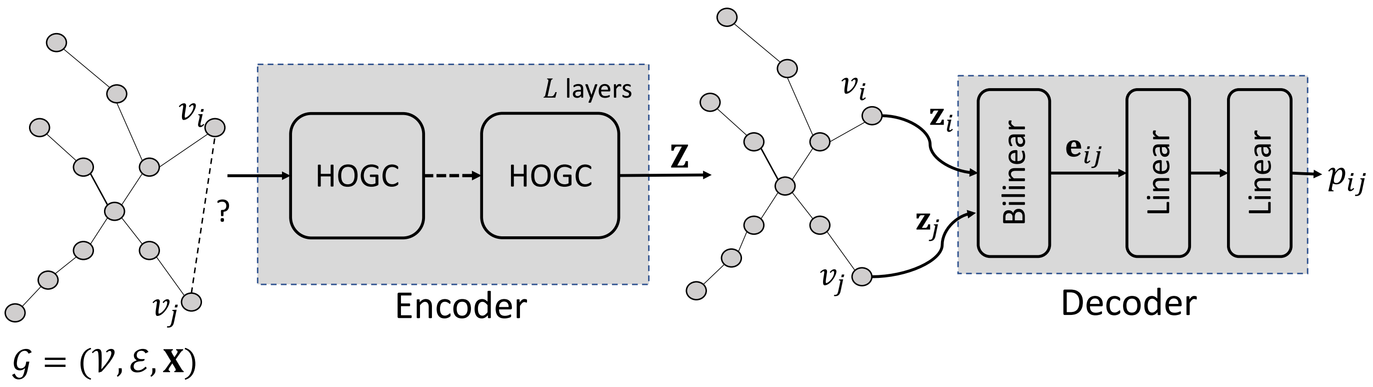


Fig. 1. Block diagram of proposed HOGCN model with L HOGC layers. Given a biomedical interaction network $\mathcal{G}$ with initial features $\mathbf{X}$ for biomedical entities, the encoder mixes the feature representation of neighbors at various distances and learn final representation $\mathbf{Z}$. The decoder takes the representation $\mathbf{z}_i$ and $\mathbf{z}_j$ of nodes v_i and v_j to learn the representation $\mathbf{e}_{ij}$ for the edge (denoted by ?) and predict probability p_{ij} of its existence.

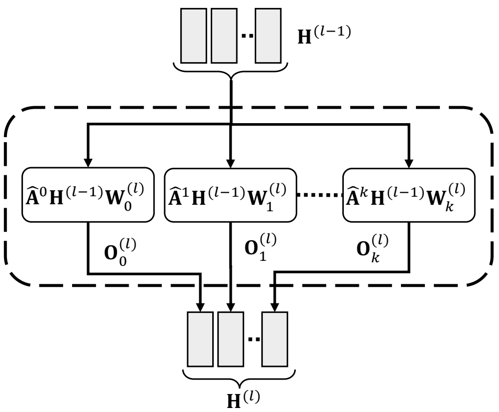

$\mathbf{H}^{(l-1)}$
$\widehat{\mathbf{A}}^0\mathbf{H}^{(l-1)}\mathbf{W}_0^{(l)}$
$\widehat{\mathbf{A}}^1\mathbf{H}^{(l-1)}\mathbf{W}_1^{(l)}$
$\widehat{\mathbf{A}}^k\mathbf{H}^{(l-1)}\mathbf{W}_k^{(l)}$
$\mathbf{O}_0^{(l)}$
$\mathbf{O}_1^{(l)}$
$\mathbf{O}_k^{(l)}$
$\mathbf{H}^{(l)}$

Fig. 2. High-order graph convolution (HOGC) Layer with $\mathbb{P} = \{0, \ldots, k\}$. The feature representation $\mathbf{H}^{(l)}$ is a linear combination of the neighbors $\widehat{\mathbf{A}}^j\mathbf{H}^{(l-1)}$ at multiple distances j. $\mathbf{O}_j^{(l)}$ represents feature representation of neighbors at j distances for layer l.

The maximum order of neighborhood k and the number of trainable weight matrices $|\mathbb{P}|$, one per each adjacency power, can vary across layers. However, we set the same k for neighborhood aggregation and the same dimension d for all the weight matrices across all layers.

Neighborhood features from different adjacency powers $j \in \{0, 1, \ldots, k\}$ at layer $(l-1)$ are column-wise concatenated to obtain feature representation $H^{(l-1)}$. As shown in Fig 2, weight $\mathbf{W}_{(\cdot)}^{(l)}$ at layer l can learn the arbitrary linear combination of the concatenated features to obtain $\mathbf{H}^{(l)}$. Specifically, the layer can assign different coefficients to different columns in the concatenated matrix. For instance, to learn feature differences between first order and second order neighbors using HOGCN with $k = 2$, the layer can assign positive coefficients to the columns produced by $\widehat{\mathbf{A}}$ and assign negative coefficients to $\widehat{\mathbf{A}}^2$ *i.e* $f(\sigma(\widehat{\mathbf{A}}\mathbf{X}) - \sigma(\widehat{\mathbf{A}}^2\mathbf{X}))$. This allows the model to learn feature differences among neighbors at various distances. However, GCN-based models fail to preserve neighborhood-specific information and cannot learn those differences [13]. We thus use L HOGC layers to learn the latent representation $\mathbf{Z} \in \mathbb{R}^{|\mathcal{V}| \times d^*}$ for biomedical entities in the network, where $d^* = d \times |\mathbb{P}|$ and d is the dimension of node's representation for each adjacency power.

4.2 Interaction Decoder

We introduced the encoder based on HOGC layers that learns feature representation $\mathbf{Z}$ for biomedical entities by mixing neighborhood information at multiple distances. Next, we discuss the decoder that reconstructs the interactions in $\mathcal{G}$ based on the representation $\mathbf{Z}$.

We adopt a bilinear layer to fuse the representation of biomedical entities v_i and v_j and learn the edge representation $\mathbf{e}_{ij}$. More precisely, we define a simple bilinear layer that takes the representation $\mathbf{z}_i \in \mathbb{R}^{d^* \times 1}$ and $\mathbf{z}_j \in \mathbb{R}^{d^* \times 1}$ as input:

$$\mathbf{e}_{ij} = \mathrm{ELU}(\mathbf{z}_i^T \mathbf{W}_b \mathbf{z}_j + \mathbf{b}) \tag{3}$$

where $\mathbf{W}_b \in \mathbb{R}^{d \times d^* \times d^*}$ represents the learnable fusion matrix, $\mathbf{e}_{ij}$ is the representation of edge e_{ij} between nodes v_i and v_j, $\mathbf{b}$ denotes the bias of the bilinear layer. ELU is non-linearity. We stack $\mathbf{z}_i^T$ d times and obtain a matrix $\in \mathbb{R}^{d \times 1 \times d^*}$. Similarly, we stack $\mathbf{z}_j$ d times to obtain a matrix $\in \mathbb{R}^{d \times d^* \times 1}$. The multiplication then results in $\mathbf{e}_{ij} \in \mathbb{R}^d$.

The edge representation $\mathbf{e}_{ij}$ is then fed into 2-layered fully connected neural network to predict probability p_{ij} for edge e_{ij}:

$$p_{ij} = \mathrm{sigmoid}(\mathrm{FC}_2(\mathrm{ELU}(\mathrm{FC}_1(\mathbf{e}_{ij})))) \tag{4}$$

So far, we have discussed the encoder and decoder of our proposed approach. Next, we describe the training procedure of our proposed HOGCN model. In particular, we explain how to optimize the trainable neural network weights in an end-to-end manner.

4.3 Training HOGCN

During HOGCN training, we employ binary cross entropy loss to optimize the model parameters

$$\mathcal{L}(v_i, v_j) = -A_{ij}\log(p_{ij}) - (1 - A_{ij})\log(1 - p_{ij}) \tag{5}$$

and encourage the model to assign higher probability to observed interactions (v_i, v_j) than to randomly selected non-interactions. p_{ij} is the predicted interaction probability between v_i and v_j and A_{ij} denotes the ground-truth interaction label between these nodes. The final loss function considering all interactions is

$$\mathcal{L} = \sum_{(i,j)\in\mathcal{E}} \mathcal{L}(v_i, v_j) \qquad (6)$$

We follow an end-to-end approach to jointly optimize over all trainable parameters and backpropagate the gradients through the encoder and decoder of HOGCN.

4.3.1 Algorithm

HOGCN leverages biomedical network structure $\mathbf{A}$ along with additional information about biomedical entities as the initial feature representation $\mathbf{X}$. In this paper, we initialize the initial features $\mathbf{X}$ to be one-hot encoding *i.e.* $\mathbf{I}_{|\mathcal{V}|}$. The feature matrix $\mathbf{X}$ can be initialized with properties of biomedical entities or pre-trained embeddings from other network-based approaches such as DeepWalk, node2vec.

Algorithm 1 Training of HOGCN for biomedical interaction prediction

```
1: Inputs: $\widehat{\mathbf{A}}$, $\mathbf{X}$, $k$
2: $\mathbf{H}^{(0)} = \mathbf{X}$
3: for $t = 1$ to $T$ do
4:     Sample mini-batch of training edges and their interaction labels
5:     for $l = 1$ to $L$ do
6:         $\mathbf{B} := \mathbf{H}^{(l-1)}$
7:         for $j = 1$ to $k$ do
8:             $\mathbf{B} := \widehat{\mathbf{A}}\mathbf{B}$
9:             $\mathbf{O}_j^{(l)} := \mathbf{B}\mathbf{W}_j^{(l)}$
10:        end for
11:        $\mathbf{H}^{(l)} := \big\Vert_{j\in\mathbb{P}} \mathbf{O}_j^{(l)}$
12:    end for
13:    $\mathbf{Z} := \mathbf{H}^{(L)}$
14:    $p_{ij} :=$ Interaction Decoder($\mathbf{Z}$)
15:    Compute loss in (6)
16:    Update model parameters via gradient descent
17: end for
```

Given an adjacency matrix $\mathbf{A}$ and the initial node representations $\mathbf{X}$, the higher-order neighborhood indicated by the higher power of the adjacency matrix is iteratively computed which makes the model more efficient. By adopting right-to-left multiplication, for instance, we can calculate $\widehat{\mathbf{A}}^3\mathbf{H}^{(i)}$ as $\widehat{\mathbf{A}}(\widehat{\mathbf{A}}(\widehat{\mathbf{A}}^{(i)}))$ (Line 8 in Algorithm 1). Representation $\mathbf{O}_j^{(l)}$ learned for the neighborhood at j distances are concatenated to obtain the representation $\mathbf{H}^{(l)}$ as shown in Fig. 2 (Line 11 in Algorithm 1). After passing through L HOGC layers, we obtain the final representation $\mathbf{Z}$ for biomedical entities. With the final representations $\mathbf{Z}$ and the mini-batch of training edges, we retrieve the embeddings for the nodes in training edges and feed them into the interaction decoder to compute their interaction probabilities.

The parameters of HOGCN are optimized with a binary cross-entropy loss (Equation (6)) in an end-to-end manner. Given two biomedical entities v_i and v_j, the trained model can predict the probability of their interactions.

5 Experimental design

We view the problem of biomedical interaction prediction as solving a link prediction task on an interaction network. We consider various interaction datasets and compare our proposed method with the state-of-the-art methods.

5.1 Datasets

We conduct interaction prediction experiments on four publicly-available biomedical network datasets:

- **BioSNAP-DTI** [28]: DTI network contains 15,139 drug-target interactions between 5,018 drugs and 2,325 proteins.
- **BioSNAP-DDI** [28]: DDI network contains 48,514 drug-drug interactions between 1,514 drugs extracted from drug labels and biomedical literature.
- **HuRI-PPI** [2]: HI-III human PPI network contains 5,604 proteins and 23,322 interactions generated by multiple orthogonal high-throughput yeast two-hybrid screens.
- **DisGeNET-GDI** [29]: GDI network consists of 81,746 interactions between 9,413 genes and 10,370 diseases curated from GWAS studies, animal models and scientific literature.

We consider binary interactions $A_{ij} \in \{0, 1\}$ for all four datasets in this paper where $A_{ij} = 0$ represents the absence of interactions and $A_{ij} = 1$ represents the presence of interactions based on some evidences. Since we only have an adjacency matrix $\mathbf{A}$ representing the interaction information, we initialize feature matrix $\mathbf{X} = \mathbf{I}$. We plan to integrate features of biological entities such as expression data for genes [20], sequence features for proteins [30] to initialize $\mathbf{X}$ in the future. Table 2 provides summary of datasets used in our experiments. We provide the number of interactions used for training, validation, and testing for each interaction datasets. Also, the table includes the average number of interactions for each biomedical entity which can be computed as $\frac{2|\mathcal{E}|}{|\mathcal{V}|}$.

5.2 Baselines

We compare our proposed model with the following network-based baselines for interaction prediction:

- network similarity-based methods
  - **L3** [14] counts the number of paths with length-3 normalized by the degree for all the node pairs.
- Network embedding methods
  - **DeepWalk** [15] performs truncated random walk exploring the network neighborhood of nodes and applies skip-gram model to learn the d-dimensional embedding for each node in the network. Node features are concatenated to form edge representation and train a logistic regression classifier.
  - **node2vec** [16] extends DeepWalk by running biased random walk based on breadth/depth-first search to capture both local and global network structure.
- Graph convolution-based methods

TABLE 2
Summary of the datasets used in our experiments.

Dataset	# Nodes	# Edges				Avg. node degree
		Training (70%)	Validation (10%)	Testing (20%)	Total	
DTI	5,018 drugs, 2,325 proteins	10,597	1,514	3,028	15,139	4.12
DDI	1,514 drugs	33,960	4,852	9,702	48,514	64.09
PPI	5,604 proteins	16,326	2,332	4,664	23,322	8.32
GDI	9,413 genes, 10,370 diseases	57,222	8,175	16,349	81,746	8.26

- **VGAE** [10] uses graph convolutional encoder with two GCN layers to learn representation for each node in the network and adopts inner product decoder to reconstruct adjacency matrix.
- **GCN** [17] uses normalized adjacency matrix to learn node representations. The representation for nodes are concatenated to form feature representation for the edges and fully connected layer use these edge representation to reconstruct edges, similar to HOGCN. Setting $\mathbb{P} = \{1\}$ in our proposed HOGCN is equivalent to GCN.
- **SkipGNN** [12] learns the node embeddings by combining direct and skip similarity between nodes. Setting $\mathbb{P} = \{1, 2\}$ in our proposed HOGCN is equivalent to SkipGNN.

5.3 Experimental setup

We split the interaction dataset into training, validation, and testing interactions in a ratio of 7:1:2 as shown in Table 2. Since the available interactions are positive samples, the negative samples are generated by randomly sampling from the complement set of positive examples. Five independent runs of the experiments with different random splits of the dataset are conducted to report the prediction performance. We use (1) area under the precision-recall curve (AUPRC) and (2) area under the receiver operating characteristics (AUROC) as the evaluation metrics. With these evaluation metrics, we expect positive interactions to have higher interaction probability compared to negative interactions. So, the higher value of AUPRC and AUROC indicates better performance.

We implement HOGCN using PyTorch [31] and perform all experiments on a single NVIDIA GeForce RTX 2080Ti GPU. We construct a 2-layered HOGC network with $k = 3$ for each layer. At each HOGC layer, the node mixes the feature representations from neighbors at distances $\mathbb{P} = \{0, 1, 2$ and $3\}$. The dimension of all weight matrices in HOGC layers is set to $d = 32$. All the weight matrices are initialized using Xavier initialization [32]. We train our model using mini-batch gradient descent with Adam optimizer [33] for a maximum of 50 epochs, with a fixed learning rate of 5×10^{-4}. We set the mini-batch size to 256 and the dropout probability [34] to 0.1 for all layers. Early stopping is adopted to stop training if validation performance does not improve for 10 epochs. The dimension of the edge feature $\mathbf{e_{ij}}$ from the bilinear layer is 64 followed by linear layers to project the edge features to edge probabilities. For baseline methods, we follow the same experimental settings discussed in [12].

6 Results

In this section, we investigate the performance and flexibility of HOGCN on interaction prediction using four different datasets. We further explore the robustness of HOGCN to sparse networks. Finally, we demonstrate the ability of HOGCN to make novel predictions with literature-based case studies.

6.1 Biomedical interaction prediction

We compare HOGCN against various baselines on biomedical interaction prediction tasks using four different types of interaction datasets including protein-protein interactions (PPIs), drug-target interactions (DTIs), drug-drug interactions (DDIs), and gene-disease associations (GDIs).

We randomly mask 20% of interactions from the network as a test set and 10% as a validation set. We train all models with 70% of interactions and evaluate their performances on test sets. The best set of hyperparameters is selected based on their performances on the validation dataset. Finally, the experiment is repeated for five independent random splits of the interaction dataset and the results with $\pm$ one standard deviation are reported in Table 3. All of our models used for the reported results are of same capacity (i.e. $\mathbb{P} = \{0, 1, 2, 3\}$ and $d = 32$).

Table 3 shows that HOGCN achieves huge improvement over network embedding methods such as DeepWalk and node2vec across all datasets. Specifically, HOGCN outperforms Deepwalk on AUPRC by 24.44% in DTI, 28.51% in DDI, 30.07% in PPI, and 13.79% in GDI. Although node2vec achieves better performance compared to DeepWalk by adopting a biased random walk, HOGCN still outperforms node2vec by a significant margin. DeepWalk and node2vec consider different orders of neighborhood defined by the window size and learns similar representations for the nodes in that window. In contrast, HOGCN learns feature differences between neighbors at various distances to obtain feature representation for the node and thus achieves superior performance. The improved performance suggests that feature differences between different order neighbors provide important information for interaction prediction.

A network similarity-based method, L3 [14] outperforms network embedding methods across four datasets but is limited to a single aspect of network similarity i.e. the number of paths of length 3 connecting two nodes. So, L3 cannot be applied when other similarities between nodes such as similarity in features and common neighbors at various distances need to be considered. HOGCN overcomes these limitations and outperforms L3 across all interaction datasets with huge gain. In particular, HOGCN gains 3.5% AUPRC and 7.09% AUROC on PPI over L3 [14], which

TABLE 3
Average AUPRC and AUROC with ± one standard deviation on biomedical interaction prediction

Dataset	Method	AUPRC	AUROC
DTI	DeepWalk	0.753 ± 0.008	0.735 ± 0.009
	node2vec	0.771 ± 0.005	0.720 ± 0.010
	L3	0.891 ± 0.004	0.793 ± 0.006
	VGAE	0.853 ± 0.010	0.800 ± 0.010
	GCN	0.904 ± 0.011	0.899 ± 0.010
	SkipGNN	0.928 ± 0.006	0.922 ± 0.004
	HOGCN	**0.937 ± 0.001**	**0.934 ± 0.001**
DDI	DeepWalk	0.698 ± 0.012	0.712 ± 0.009
	node2vec	0.801 ± 0.004	0.809 ± 0.002
	L3	0.860 ± 0.004	0.869 ± 0.003
	VGAE	0.844 ± 0.076	0.878 ± 0.008
	GCN	0.856 ± 0.005	0.875 ± 0.004
	SkipGNN	0.866 ± 0.006	0.886 ± 0.003
	HOGCN	**0.897 ± 0.003**	**0.911 ± 0.002**
PPI	DeepWalk	0.715 ± 0.008	0.706 ± 0.005
	node2vec	0.773 ± 0.010	0.766 ± 0.005
	L3	0.899 ± 0.003	0.861 ± 0.003
	VGAE	0.875 ± 0.004	0.844 ± 0.006
	GCN	0.909 ± 0.002	0.907 ± 0.006
	SkipGNN	0.921 ± 0.003	0.917 ± 0.004
	HOGCN	**0.930 ± 0.002**	**0.922 ± 0.001**
GDI	DeepWalk	0.827 ± 0.007	0.832 ± 0.003
	node2vec	0.828 ± 0.006	0.834 ± 0.003
	L3	0.899 ± 0.001	0.832 ± 0.001
	VGAE	0.902 ± 0.006	0.873 ± 0.009
	GCN	0.909 ± 0.002	0.906 ± 0.006
	SkipGNN	0.915 ± 0.003	0.912 ± 0.004
	HOGCN	**0.941 ± 0.001**	**0.936 ± 0.001**

recently outperformed 20 network science methods in the PPI prediction problem.

Graph convolution-based methods such as GCN and VGAE achieves significant improvements over network embedding approaches but achieves comparable performance with L3. SkipGNN shows improvement over all other methods by incorporating skip similarity to aggregate information from second-order neighbors. Moreover, HOGCN with $k = 3$ achieves an improvement over all graph convolution-based methods. Specifically, HOGCN achieves improvement in AUPRC over VGAE [10] by 6.3%, GCN [17] by 4.8% and SkipGNN [12] by 3.6% on DDI dataset.

As HOGCN can learn the linear combination of node features at multiple distances, it can extract meaningful representations from the interaction networks. The results in Table 3 demonstrate that our approach with higher-order neighborhood mixing outperforms the state-of-the-art methods on real interaction datasets.

6.2 Exploration of HOGCN's drug representations

Next, we evaluate if HOGCN learns meaningful representation when feature representations of higher-order neighbors are aggregated. To this aim, we train GCN, SkipGNN, and HOGCN models on the DDI network with training/validation/testing split provided in Table 2 to obtain the drug representations $\mathbf{Z}$. The learned drug representations are mapped to 2D space using t-SNE [35] and visualize them in Fig. 3.

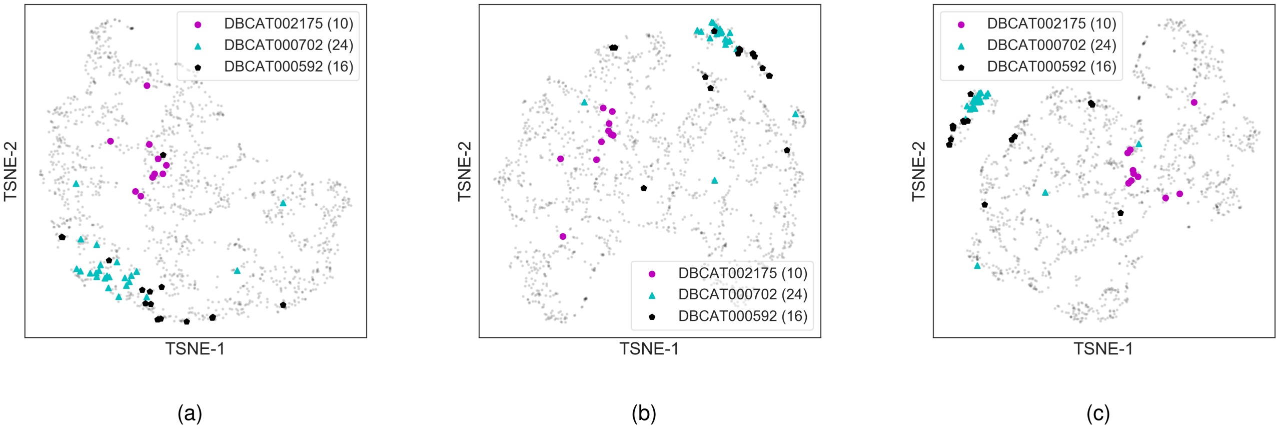
Drugbank [36] provides information about drugs and their categories based on different characteristics such as involved metabolic enzymes, class of drugs, side effects of drugs, and the like. For this experiment, we collect drug categories from Drugbank and limit the selection of drug categories such that the training set doesn't contain any interactions between the drugs in the same category. The selected drug categories are ACE Inhibitors and Diuretics (DBCAT002175), Penicillins (DBCAT000702), and Antineoplastic Agents (DBCAT000592) with 10, 24, and 16 drugs respectively. Although these drugs don't have direct interactions in the training set, we assume that these drugs share neighborhoods at various distances and can be explored accordingly with HOGCN.

Fig. 3 shows the clustering structure in drugs' representations as neighborhood information at multiple distances are considered. Examining the figure, we observe that drugs in the same category are embedded close to each other in the 2D space when the model aggregates information from farther neighbors. For example, 24 drugs in the Penicillins (DBCAT000702) category (marked with blue triangles in Fig. 3) are scattered in the representation space learned by GCN that only considers feature aggregation from immediate neighbors (Fig. 3a). Note that these drugs don't have any direct interaction between themselves in the training set. Since GCN-based models can only average the representation from immediate neighbors, these drugs are mapped relatively farther to each other and closer to other interacting drugs. SkipGNN considers skip similarity to aggregate features from second-order neighbors and show relatively compact clusters compared to GCNs (Fig. 3b). On the other hand, HOGCN considers the higher-order neighborhood and learns similar representations for drugs that belong to the same category demonstrated by compact clustering structure in Fig. 3c even though no information about categorical similarity is provided to the model. This analysis demonstrates that HOGCN learns meaningful representation for drugs by aggregating feature representations from the neighborhood at various distances.

Next, we test if the clustering pattern in Fig. 3 holds across many drug categories. With this aim, we consider all drug categories in DrugBank and compute the average Euclidean distance between each drug's representation and representations of other drugs within the same drug category. We then perform 2-sample Kolmogorov–Smirnov tests and found that HOGCN learns significantly more similar representations of drugs than expected by chance (p-value = $4.93e - 106$), GCNs (p-value = $5.05e - 56$) and SkipGNN ($1.29e - 12$). Thus, this analysis indicates that HOGCN learns meaningful representations for drugs by aggregating neighborhood information at various distances.

6.3 Robustness to network sparsity

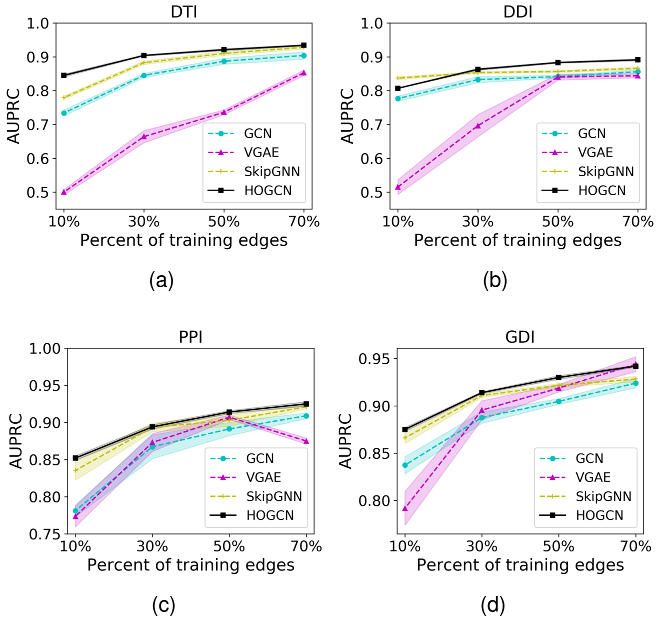
We next explore the robustness of network-based interaction prediction models to network sparsity. To this aim, we evaluate the performance with respect to the percentage of training edges varying from 10% to 70%. We make predictions on the rest of the interactions. We further use 10% of test edges for validation to select the best hyperparameter settings. For a fair comparison, we compare with



(a) (b) (c)

Fig. 3. Visualization of learned representation for drugs with (a) GCN (b) SkipGNN (c) HOGCN. Drugs are mapped to the 2D space using t-SNE package [35] with learned drug representations. Drugs categories such as DBCAT002175, DBCAT000702 and DBCAT000592 are highlighted. The number of drugs in each categroy is reported in legend. Best viewed on screen.

graph convolution-based methods that aggregate information from direct and/or second-order neighbors.



(a) (b)

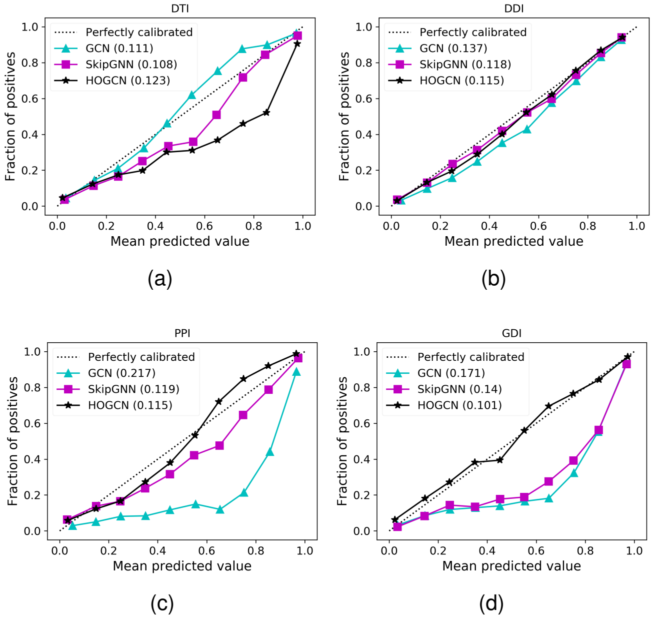
(c) (d)

Fig. 4. AUPRC comparison of HOGCN's performance with that of alternative approaches with respect to network sparsity. HOGCN consistently achieves better performance in various fraction of training edges.

Fig. 4 shows the robustness of HOGCN to network sparsity. HOGCN achieves strong performance in all tasks with different network sparsity. The performance of HOGCN steadily improves with the increase in training edges. The mixing of features from a higher-order neighborhood in HOGCN and SkipGNN shows improvement over GCN and VGAE that only consider direct neighbors. Since HOGCN can learn the linear combination of features from a 3-hop neighborhood for this experiment, it shows improvement over SkipGNN in almost all cases. This demonstrates that features from farther distances are informative for interaction prediction in sparse networks.

6.4 Calibrating model's prediction

All graph convolution-based model proposed for biomedical link prediction predicts the confidence estimate p_{ij} for interaction between two biomedical entities v_i and v_j. We thus test if a predicted confidence p_{ij} represents the likelihood of being true interaction. In other words, we expect the confidence estimate p_{ij} to be calibrated, *i.e.* p_{ij} represents true interaction probability [37]. For example, given 100 predictions, each with the confidence of 0.9, we expect that 90 interactions should be correctly classified as true interactions.



(a) (b)

(c) (d)

Fig. 5. Reliability diagrams for different graph convolution-based methods. The calibration performance is evaluated with Brier score, reported in the legend (lower is better).

To evaluate the calibration performance, we use reliability diagrams [37] and Brier score [38]. In particular, the reliability diagram provides a visual representation of model calibration. These diagrams plot the expected fraction of positives as a function of predicted probability [37]. A model with perfectly calibrated predictions is represented by a diagonal in Fig. 5. In addition to reliability diagrams, it is more convenient to have scalar summary statistics of calibration. Brier score [38] is a proper scoring rule for measuring the accuracy of predicted probabilities. A lower Brier score indicates better calibration of a set of predictions. It is computed as the mean squared error of a predicted probability p_{ij} and the ground-truth interaction label A_{ij}. Mathematically, the Brier score can be computed as:

$$\text{Brier score} = \frac{1}{|\mathcal{E}'|} \sum_{(i,j)=1}^{|\mathcal{E}'|} (p_{ij} - A_{ij})^2 \tag{7}$$

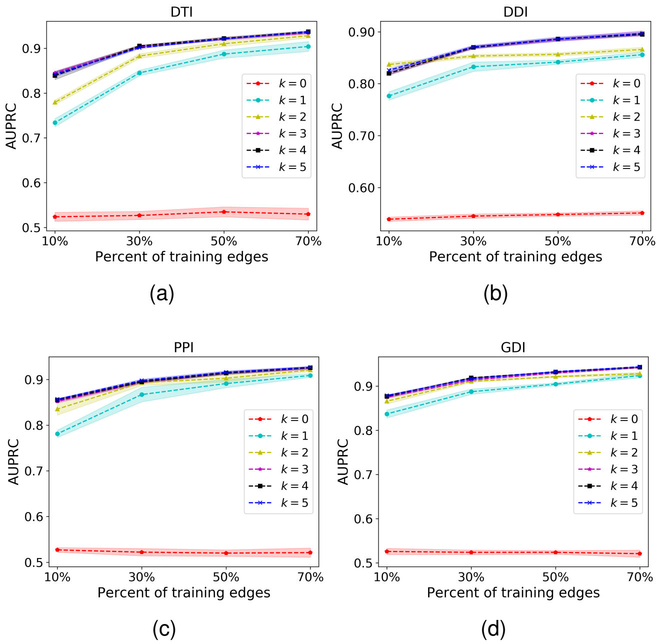
where $|\mathcal{E}'|$ denotes the number of test edges.

Fig. 5 shows the calibration plots for GCN, SkipGNN and HOGCN ($k = 3$). For the DTI dataset, SkipGNN shows better calibration compared to GCN and HOGCN (Fig. 5a), indicating that second-order neighborhood information is appropriate and aggregating features from farther away makes the model overconfident. For other datasets, GCNs are relatively overconfident for all predicted confidence. For example, approximately $20\% - 30\%$ of interactions are true positives among the interactions with high predicted confidence 0.8 in PPI (Fig. 5c) and GDI dataset (Fig. 5d). In contrast, HOGCN achieves a lower Brier score in comparison to the GCN and SkipGNN across DDI, PPI, and GDI datasets, alluding to the benefits of aggregating higher-order neighborhood features for calibrated prediction. This analysis demonstrates that HOGCN with higher-order neighborhood mixing makes accurate and calibrated predictions for biomedical interaction.

6.5 Impact of higher-order neighborhood mixing

In Section 6.3, we contrast HOGCN's performance with that of alternative graph convolution-based methods in varying fraction of edges. In this experiment, we aim to observe the performance of HOGCN when the order k is increased to allow the model to aggregate neighborhood information from farther away. We follow a similar setup as discussed in 6.3.

Fig. 6 shows the comparison of HOGCN with higher-order neighborhood mixing k such that $\mathbb{P} = \{0, 1, 2, 3, 4, 5\}$. The prediction performance of HOGCN improves with the increase in the number of training interactions for all cases except $k = 0$. For HOGCN with $k = 0$, the model only considers the features of the biological entities $i.e.$ identity matrix I and performs worse since the identity matrix is not informative about interactions. The results show that HOGCN's performances are not sensitive to the hyperparameter settings of k for all datasets since for settings $\mathbb{P} = \{3, 4, 5\}$, there is no clear differences and achieve comparable performances across the datasets. This analysis indicates that the 3-hop neighborhood ($k = 3$) provides sufficient information for interaction prediction across all datasets and the performance remains stable even with a large value for k.



(a)

(b)

(c)

(d)

Fig. 6. AUPRC comparison for higher-order message passing with different fractions of training edges. The values of k for different HOGCN models are reported in the legend.

6.6 Investigation of novel predictions

Next, we perform the literature-based validation of novel predictions. Our goal is to evaluate the quality of HOGCN's novel predictions compared to that of GCN and SkipGNN and show that HOGCN predicts novel interactions with higher confidence. For this evaluation, we consider DDIs and GDIs that have readily available databases.

We first evaluate the potential of the HOGCN to make novel GDI predictions. We collect 1,134,942 GDIs and their scores from DisGeNET [29]. The score corresponds to the number and types of sources and the number of publications supporting the associations. With the score threshold of 0.18, we obtain 17,893 new GDIs that are not in the training set. We make predictions on these 17,893 GDIs with GCN, SkipGNN, and HOGCN ($k = 3$). Out of 17,893 GDIs, HOGCN predicts a higher probability than GCN for 17,356 (96.99%) GDIs and SkipGNN for 11,418 (63.8%) GDIs. Table 4 shows the top 5 GDIs with a significant increase in interaction probabilities when higher-order neighborhood mixing is considered. We also provide the number of evidence from DisGenNet [29] to support these predictions. Improvement in predicted probabilities by HOGCN models shows that aggregating feature representations from higher-order neighbors make HOGCN more confident about the potential interactions as discussed in Section 6.4.

We select two predicted GDIs with a large number of supporting evidence and investigate the reason for the improvement in predicted confidence with HOGCN. Specifically, we choose gene-disease pairs (a) ABO and Pancreatic carcinoma (26 pieces of evidence) and (b) GPC3 and Hepatoblastoma ((17 pieces of evidence). To explain the prediction, the subnetworks containing all shortest paths between these pairs are selected. In particular, there are 49 shortest paths of length 3 between ABO and Pancreatic carcinoma including

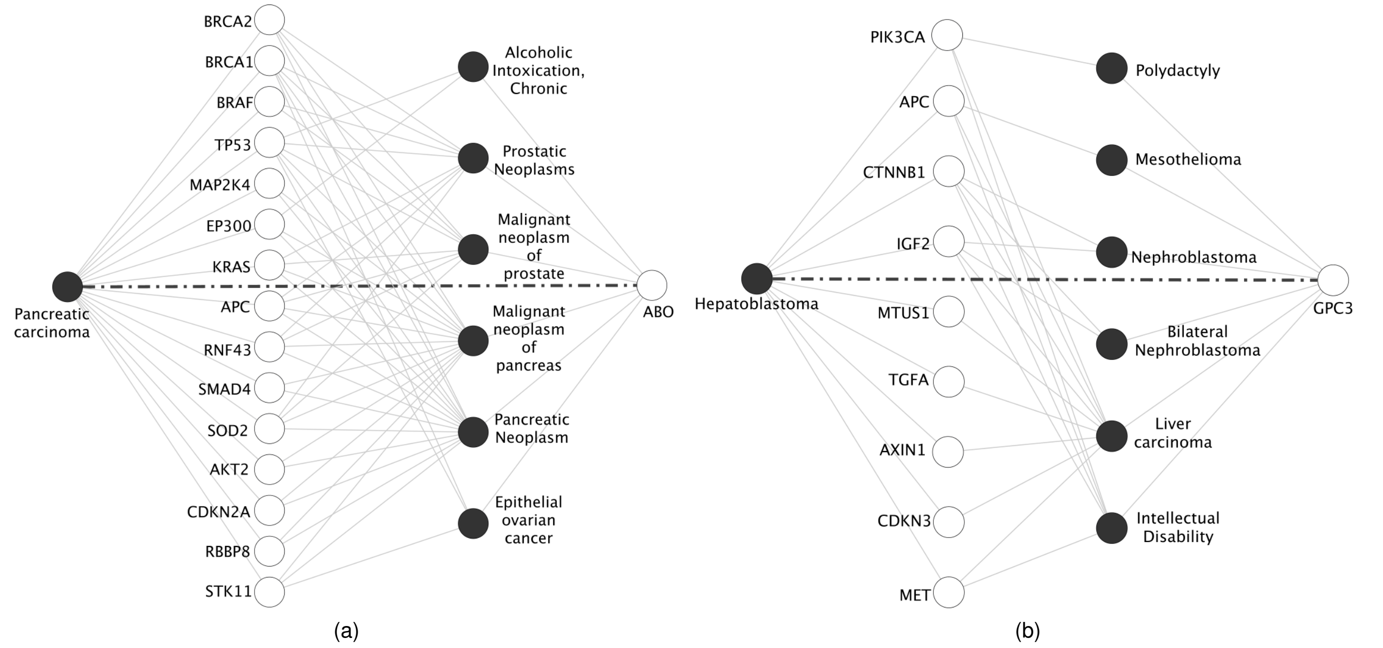


Fig. 7. Subnetwork with predicted interactions (marked by bold dashed lines) between (a) ABO and Pancreatic carcinoma (b) GPC3 and Hepatoblastoma and all shortest paths between these pairs. The known interactions are presented as gray lines. Diseases are presented as dark circles and genes are presented as white circles.

TABLE 4
Novel prediction of GDIs with the number of evidence from DisGenNet [29] supporting the interaction. GCN, SkipGNN and HOGCN are denoted by $1, 2$ and 3 respectively.

		Probability			No. of
Gene	Disease	1	2	3	Evidence
PTGER1	Gastric ulcer	0.087	0.519	0.721	1
ANGPT1	Gastric ulcer	0.173	0.583	0.657	2
ABO	Pancreatic carcinoma	0.265	0.615	0.770	26
VCAM1	Endotoxemia	0.294	0.529	0.639	5
GPC3	Hepatoblastoma	0.307	0.540	0.598	17

6 diseases and 15 genes (Fig. 7a). Similarly, there are 20 shortest paths of length 3 between GPC3 and Hepatoblastoma including 6 diseases and 9 genes (Fig. 7b). Since these nodes are 3-hop away from each other and GCNs can only consider immediate neighbors, GCNs assign low confidence to these interactions.

Examining the subnetwork in Fig. 7a, we found that most of the diseases are related to a cancerous tumor in the pancreas and the prostate. Furthermore, pancreatic carcinoma is associated with other diseases such as Pancreatic neoplasm, malignant neoplasm of pancreas, and malignant neoplasm of prostate [29]. Since ABO is linked with diseases that are related to pancreatic carcinoma and other genes are related to these diseases as well, HOGCN captures such association (Fig 7a) even though they are farther away in the network. Similarly, HOGCN predicts association for GPC3 and Hepatoblastoma (Fig. 7b).

Next, we perform a similar case study for DDIs and evaluate the predictions against DrugBank [36]. For this experiment, we predict every drug pair with GCN, SkipGNN, and HOGCN and exclude the interactions that are already in the training set. Table 5 shows the top 5 interactions with an increase in interaction probabilities when higher orders of the neighborhood are considered. As discussed in Section 6.4, HOGCN makes predictions with higher confidence compared to GCN and SkipGNN for the interactions that are likely to be a true positive.

TABLE 5
Novel prediction of DDIs with the literature evidence supporting the interaction. GCN, SkipGNN and HOGCN are denoted by $1, 2$ and 3 respectively.

Drug 1	Drug 2	Probability			Evidence
		1	2	3	
Nelfinavir	Acenocoumarol	0.192	0.318	0.417	[39]
Praziquantel	Itraconazole	0.609	0.721	0.811	[40]
Cisapride	Droperidol	0.618	0.725	0.823	[41]
Dapsone	Warfarin	0.632	0.720	0.885	[42]
Levofloxacin	Tobramycin	0.663	0.760	0.823	[43]

Moreover, we validate the false-positive DDI predictions of GCNs and investigate the subnetwork for these drugs in DDI networks to reason the predictions. Table 6 shows the top 5 interactions with a significant decrease in predicted confidence compared to GCN-based models. Since these DDIs are false positives [36], GCN-based models make overconfident predictions for such DDIs. In contrast, HOGCN significantly reduces the predicted confidence for these DDIs to be true positive, indicating that the higher-order neighborhood allows HOGCN to identify false-positive predictions. In particular, HOGCN can identity false-positive DDI between Belimumab and Estazolam even though they are 3-hop away from each other.

We select a subnetwork involving the drugs to investigate the reason for such predictions. Fig. 8 shows the subnetwork with all shortest paths between the drugs in Table 6. Examining the figure, we observe that the drugs in these false-positive DDIs have common immediate neighbors for all cases. GCN makes wrong predictions for these

TABLE 6
Predicted probability for negative DDIs. GCN, SkipGNN and HOGCN are denoted by $1, 2$ and 3 respectively.

Drug 1	Drug 2	Probability with k		
		1	2	3
Tranylcypromine	Melphalan	0.925	0.478	0.065
Belimumab	Estazolam	0.912	0.477	0.178
Methotrimeprazine	Cloxacillin	0.907	0.406	0.065
Hydrocodone	Melphalan	0.905	0.193	0.012
Ibrutinib	Mecamylamine	0.899	0.398	0.353

DDIs with high confidence. However, SkipGNN becomes less confident about the interaction being true positive by considering the skip similarity. HOGCN further reduces the predicted confidence for Tranylcypromine and Melphalan to 0.065, indicating that there is no association between these drugs.

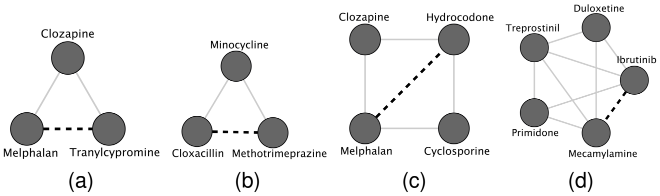


Fig. 8. Subnetwork containing false positive predictions (marked by dark dashed lines) and all shortest paths between (a) Tranylcypromine and Melphalan (b) Methotrimeprazine and Cloxacillin (c) Hydrocodone and Melphalan and (d) Ibrutinib and Mecamylamine. Other known interactions are presented as gray gray lines. Dark circles denotes drugs.

These case studies show that HOGCN with higher-order neighborhood mixing not only provide information for the identification of novel interactions but also help HOGCN to reduce false-positive predictions.

7 Conclusion

We present a novel deep graph convolutional network (HOGCN) for biomedical interaction prediction. Our proposed model adopts a higher-order graph convolutional layer to learn to mix the feature representation of neighbors at various scales. Experimental results on four interaction datasets demonstrate the superior and robust performance of the proposed model. Furthermore, we show that HOGCN makes accurate and calibrated predictions by considering higher-order neighborhood information.

There are several directions for future study. Our approach only considers the known interactions to flag potential interactions. There are other sources of biomedical information such as various physicochemical and biological properties of biomedical entities that can provide additional information about the interaction and we plan to investigate the integration of such features into the model [20], [30]. As HOGCN aggregates the neighborhood information at various distances and can flag novel interactions, it would be interesting to provide interpretable explanations for the predictions in the form of a small subgraph of the input interaction network $\mathcal{G}$ that are most influential for the predictions [44].

Acknowledgments

This work was supported by the NSF [NSF-1062422 to A.H.], [NSF-1850492 to R.L.] and the NIH [GM116102 to F.C.]

References

[1] L. Cowen, T. Ideker, B. J. Raphael, and R. Sharan, "Network propagation: a universal amplifier of genetic associations," *Nature Reviews Genetics*, vol. 18, no. 9, p. 551, 2017.

[2] K. Luck, D.-K. Kim, L. Lambourne, K. Spirohn, B. E. Begg, W. Bian, R. Brignall, T. Cafarelli, F. J. Campos-Laborie, B. Charloteaux *et al.*, "A reference map of the human binary protein interactome," *Nature*, vol. 580, no. 7803, pp. 402–408, 2020.

[3] M. Zitnik, M. Agrawal, and J. Leskovec, "Modeling polypharmacy side effects with graph convolutional networks," *Bioinformatics*, vol. 34, no. 13, pp. i457–i466, 2018.

[4] Y. Luo, X. Zhao, J. Zhou, J. Yang, Y. Zhang, W. Kuang, J. Peng, L. Chen, and J. Zeng, "A network integration approach for drug-target interaction prediction and computational drug repositioning from heterogeneous information," *Nature communications*, vol. 8, no. 1, pp. 1–13, 2017.

[5] M. Agrawal, M. Zitnik, J. Leskovec *et al.*, "Large-scale analysis of disease pathways in the human interactome." in *PSB*. World Scientific, 2018, pp. 111–122.

[6] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: going beyond euclidean data," *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 18–42, 2017.

[7] J. Qiu, J. Tang, H. Ma, Y. Dong, K. Wang, and J. Tang, "Deepinf: Social influence prediction with deep learning," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, p. 2110–2119.

[8] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, p. 974–983.

[9] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 2017, pp. 1263–1272.

[10] T. N. Kipf and M. Welling, "Variational graph auto-encoders," *arXiv preprint arXiv:1611.07308*, 2016.

[11] X. Yue, Z. Wang, J. Huang, S. Parthasarathy, S. Moosavinasab, Y. Huang, S. M. Lin, W. Zhang, P. Zhang, and H. Sun, "Graph embedding on biomedical networks: Methods, applications and evaluations," *Bioinformatics*, vol. 36, no. 4, pp. 1241–1251, 2020.

[12] K. Huang, C. Xiao, L. Glass, M. Zitnik, and J. Sun, "Skipgnn: Predicting molecular interactions with skip-graph networks," *arXiv preprint arXiv:2004.14949*, 2020.

[13] S. Abu-El-Haija, B. Perozzi, A. Kapoor, N. Alipourfard, K. Lerman, H. Harutyunyan, G. Ver Steeg, and A. Galstyan, "Mixhop: Higher-order graph convolutional architectures via sparsified neighborhood mixing," in *International Conference on Machine Learning*, 2019, pp. 21–29.

[14] I. A. Kovács, K. Luck, K. Spirohn, Y. Wang, C. Pollis, S. Schlabach, W. Bian, D.-K. Kim, N. Kishore, T. Hao *et al.*, "Network-based prediction of protein interactions," *Nature communications*, vol. 10, no. 1, pp. 1–8, 2019.

[15] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 701–710.

[16] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.

[17] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[18] I. Ahmad, M. U. Akhtar, S. Noor, and A. Shahnaz, "Missing link prediction using common neighbor and centrality based parameterized algorithm," *Scientific Reports*, vol. 10, no. 1, pp. 1–9, 2020.

[19] O. Keskin, N. Tuncbag, and A. Gursoy, "Predicting protein–protein interactions from the molecular to the proteome level," *Chemical reviews*, vol. 116, no. 8, pp. 4884–4909, 2016.

[20] K. KC, R. Li, F. Cui, Q. Yu, and A. R. Haake, "Gne: a deep learning framework for gene network inference by aggregating biological information," *BMC systems biology*, vol. 13, no. 2, p. 38, 2019.

[21] O. Shchur, M. Mumme, A. Bojchevski, and S. Günnemann, "Pitfalls of graph neural network evaluation," *arXiv preprint arXiv:1811.05868*, 2018.

[22] A. R. Benson, R. Abebe, M. T. Schaub, A. Jadbabaie, and J. Kleinberg, "Simplicial closure and higher-order link prediction," *Proceedings of the National Academy of Sciences*, vol. 115, no. 48, pp. E11 221–E11 230, 2018.

[23] S. Cao, W. Lu, and Q. Xu, "Grarep: Learning graph representations with global structural information," in *Proceedings of the 24th ACM international on conference on information and knowledge management*, 2015, pp. 891–900.

[24] D. Zhu, P. Cui, Z. Zhang, J. Pei, and W. Zhu, "High-order proximity preserved embedding for dynamic networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 11, pp. 2134–2144, 2018.

[25] R. A. Rossi, N. K. Ahmed, and E. Koh, "Higher-order network representation learning," in *Companion Proceedings of the The Web Conference 2018*, ser. WWW '18. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee, 2018, p. 3–4. [Online]. Available: https://doi.org/10.1145/3184558.3186900

[26] R. A. Rossi, N. K. Ahmed, E. Koh, S. Kim, A. Rao, and Y. A. Yadkori, "Hone: higher-order network embeddings," *arXiv preprint arXiv:1801.09303*, 2018.

[27] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, "A comprehensive survey on graph neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[28] M. Zitnik, S. Rok Sosic, and J. Leskovec, "Biosnap datasets: Stanford biomedical network dataset collection," *http://snap.stanford.edu/biodata*, 2018.

[29] J. Piñero, J. M. Ramírez-Anguita, J. Saüch-Pitarch, F. Ronzano, E. Centeno, F. Sanz, and L. I. Furlong, "The disgenet knowledge platform for disease genomics: 2019 update," *Nucleic acids research*, vol. 48, no. D1, pp. D845–D855, 2020.

[30] K. KC, F. Cui, A. Haake, and R. Li, "Interpretable structured learning with sparse gated sequence encoder for protein-protein interaction prediction," in *Proceedings of the 25th International Conference on Pattern Recognition, ICPR*, 2020.

[31] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in neural information processing systems*, 2019, pp. 8026–8037.

[32] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.

[33] D. P. Kingma and J. Ba, "Adam (2014), a method for stochastic optimization," in *Proceedings of the 3rd International Conference on Learning Representations (ICLR), arXiv preprint arXiv*, vol. 1412, 2015.

[34] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[35] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[36] D. S. Wishart, C. Knox, A. C. Guo, S. Shrivastava, M. Hassanali, P. Stothard, Z. Chang, and J. Woolsey, "Drugbank: a comprehensive resource for in silico drug discovery and exploration," *Nucleic acids research*, vol. 34, no. suppl_1, pp. D668–D672, 2006.

[37] A. Niculescu-Mizil and R. Caruana, "Predicting good probabilities with supervised learning," in *Proceedings of the 22nd international conference on Machine learning*, 2005, pp. 625–632.

[38] G. W. Brier, "Verification of forecasts expressed in terms of probability," *Monthly weather review*, vol. 78, no. 1, pp. 1–3, 1950.

[39] B. Garcia, P. De Juana, T. Bermejo, J. Gómez, P. Rondon, I. Garcia Plaza *et al.*, "Sequential interaction of ritonavir and nelfinavir with acenocoumarol (abstract 1069)," in *Seventh European Conference on Clinical Aspects and Treatment of HIV Infection. Lisbon, Portugal*, 1999.

[40] E. Perucca, "Clinically relevant drug interactions with antiepileptic drugs," *British journal of clinical pharmacology*, vol. 61, no. 3, pp. 246–255, 2006.

[41] E. L. Michalets and C. R. Williams, "Drug interactions with cisapride," *Clinical pharmacokinetics*, vol. 39, no. 1, pp. 49–75, 2000.

[42] T. Truong and J. Haley, "Probable warfarin and dapsone interaction," *Clinical and Applied Thrombosis/Hemostasis*, vol. 18, no. 1, pp. 107–109, 2012.

[43] B. Ozbek and G. Otuk, "Post-antibiotic effect of levofloxacin and tobramycin alone or in combination with cefepime against pseudomonas aeruginosa," *Chemotherapy*, vol. 55, no. 6, pp. 446–450, 2009.

[44] Z. Ying, D. Bourgeois, J. You, M. Zitnik, and J. Leskovec, "Gnnexplainer: Generating explanations for graph neural networks," in *Advances in neural information processing systems*, 2019, pp. 9244–9255.

Kishan KC received his BE degree in Computer Engineering from Tribhuwan University, Nepal, in 2014. He is a Ph.D. student in the Golisano College of Computing and Information Sciences, Rochester Institute of Technology, USA. His research interests include deep learning, heterogeneous data integration, and computational biology.

Rui Li received his BS degree from Harbin Institute of Technology, China, in 2003. He received a Ph.D. degree from Rochester Institute of Technology in 2013. He is an assistant professor in the Golisano College of Computing and Information Sciences at Rochester Institute of Technology. His research focus is on computational biology, machine learning, and artificial intelligence.

Feng Cui received his Ph.D. degree from Iowa State University. He is an associate professor in the Thomas H. Gosnell School of Life Sciences. His research interests include cancer biology, computational biology, and machine learning.

Anne R. Haake received her Ph.D. degree from the University of South Carolina and post-doctoral training at the University of Rochester School of Medicine & Dentistry. She is professor and dean of the Golisano College of Computing & Information Sciences at RIT. Her research interests include developmental, cell, and systems biology, bio/medical informatics, computational modeling, and human-computer interaction.