Self-Explainable Graph Neural Networks for Link PredictionI Network[1]

reporter: Jiale Liu

paper author:Huaisheng Zhu, Dongsheng Luo, Xianfeng Tang, Junjie Xu, Hui Liu, Suhang Wang

ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY

May 11, 2024

Catalogs

Introduction

Problems to Solve

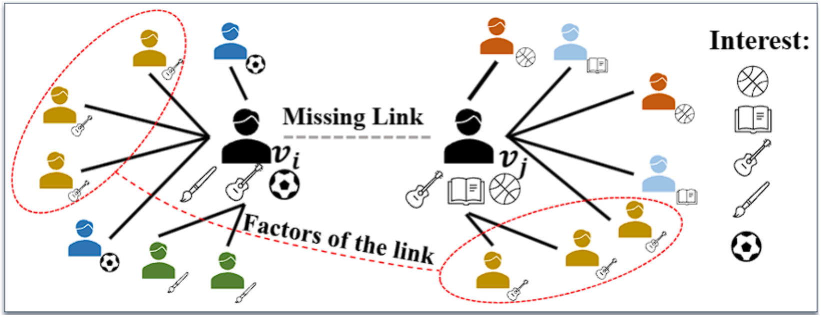
Methodology

Prediction

Loss Function

Refrence

illustration



Catalogs

Introduction

Problems to Solve

Methodology

Prediction

Loss Function

Refrence

Problems

1. for a pair of nodes (v_i, v_j), how we can identify K most important neighbors of v_j and v_i.
2.How to take both graph structure and node attributes into consideration when measuring node similarity for identifying important neighbors for link prediction?

Catalogs

Introduction

Problems to Solve

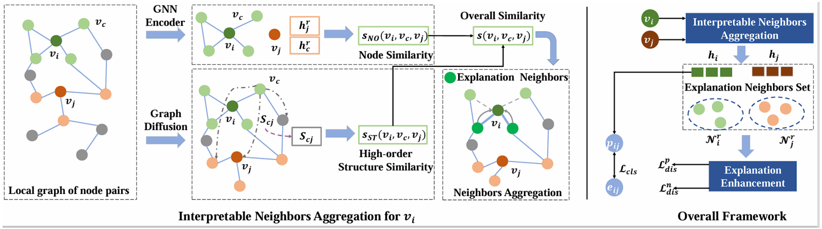
Methodology

Prediction

Loss Function

Refrence

Overview



Local graph of node pairs
v_c
v_i
v_j
GNN Encoder
Graph Diffusion
h_j^r
h_c^r
$s_{NO}(v_i, v_c, v_j)$
Node Similarity
S_{cj}
$s_{ST}(v_i, v_c, v_j)$
High-order Structure Similarity
Overall Similarity
$s(v_i, v_c, v_j)$
Explanation Neighbors
Neighbors Aggregation
Interpretable Neighbors Aggregation for v_i
Interpretable Neighbors Aggregation
h_i
h_j
Explanation Neighbors Set
$\mathcal{N}_i^r$
$\mathcal{N}_j^r$
p_{ij}
e_{ij}
$\mathcal{L}_{cls}$
$\mathcal{L}_{dis}^p$
$\mathcal{L}_{dis}^n$
Explanation Enhancement
Overall Framework

High-order Structure Similarity

To measure this high-order similarity, we propose to use a Graph Diffusion matrix which calculates the closeness of nodes in the graph structure by repeatedly passing the weighting coefficients to the neighboring nodes and represents the high-order similarity between nodes based on graph structure:

$$\mathbf{S} = \sum_{k=0}^{\infty} \theta_k \mathbf{T}^k, \tag{1}$$

$\mathbf{T}$ represents the random walk transition matrix as $\mathbf{T} = \mathbf{A}\mathbf{D}^{-1}$
PPR chooses $\theta_k^{\mathrm{PPR}} = \gamma(1-\gamma)^k$ with teleport probability $\gamma \in (0,1)$. γ is set as 0.05 in the experiment.
$\tilde{\mathbf{S}} = \mathbf{D}_S^{-1/2}\mathbf{S}\mathbf{D}_S^{-1/2}$

$$s_{\mathrm{ST}}(v_i, v_c, v_j) = \tilde{S}_{cj}, \tag{2}$$

Node Similarity

$$\mathbf{H}^m = \mathrm{MLP}(\mathbf{X}), \quad \mathbf{H}^r = \sigma(\tilde{\mathbf{A}}[\mathbf{H}^m \| \mathbf{X}]\mathbf{W}) + \mathbf{H}^m, \tag{3}$$

$$s_{\mathrm{NO}}(v_i, v_c, v_j) = \mathrm{sigmoid}\big((\mathbf{h}_j^r)^T \mathbf{h}_c^r\big), \ \forall \ v_c \in \mathcal{N}_i \tag{4}$$

$$s(v_i, v_c, v_j) = \alpha \cdot s_{\mathrm{ST}}(v_i, v_c, v_j) + (1-\alpha) \cdot s_{\mathrm{NO}}(v_i, v_c, v_j), \tag{5}$$

$$b_{ic} = \frac{\exp\left(s\left(v_i, v_c, v_j\right)\right)}{\sum_{v_c \in \mathcal{N}_i^r} \exp\left(s\left(v_i, v_c, v_j\right)\right)}. \tag{6}$$

Finally, node v_i's representation vectors can be obtained as:

$$\mathbf{h}_i = \mathbf{h}_i^r + \beta \sum_{v_c \in \mathcal{N}_i^r} b_{ic} \mathbf{h}_c^r, \tag{7}$$

Catalogs

Introduction

Problems to Solve

Methodology

Prediction

Loss Function

Refrence

Prediction

$$p_{ij} = \text{sigmoid}(\mathbf{h}_i^T \mathbf{h}_j). \tag{8}$$

Catalogs

Introduction

Problems to Solve

Methodology

Prediction

Loss Function

Refrence

Explanation Enhancement

$$\mathbf{h}_i^{\text{rand}} = \mathbf{h}_i^r + \beta \sum_{v_c \in \mathcal{N}_i^{\text{rand}}} b_{ic}^{\text{rand}} \mathbf{h}_c^r, \tag{9}$$

$$p_{ij}^{\text{rand}} = \text{sigmoid}((\mathbf{h}_i^{\text{rand}})^T \mathbf{h}_j^{\text{rand}}). \tag{10}$$

$$\mathcal{L}_{\text{dis}}^p = \sum_{e_{ij} \in \mathcal{E}_L, e_{ij}=1} \max(0, p_{ij}^{\text{rand}} + \delta - p_{ij}), \tag{11}$$

Loss

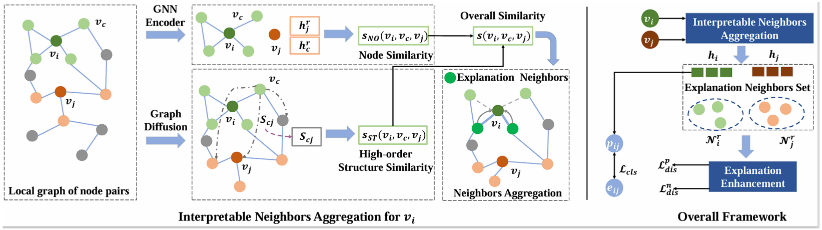
If nodes v_i and v_j have lower similarity scores with nodes in $\mathcal{N}_j^r$ and $\mathcal{N}_j^r$ respectively, the similarity of $\mathbf{h}_i$ and $\mathbf{h}_j$ will be small and the model will give a lower probability for the link of (v_i, v_j). To achieve this purpose, we randomly sample unlinked pairs $e_{ij} = 0$ which have the same number as the number of linked pairs $e_{ij} = 1$ in $\mathcal{E}_L$. The set of randomly selected unlinked pairs can be denoted as $\mathcal{E}_N$. The similarity scores can be minimized through the following loss function:

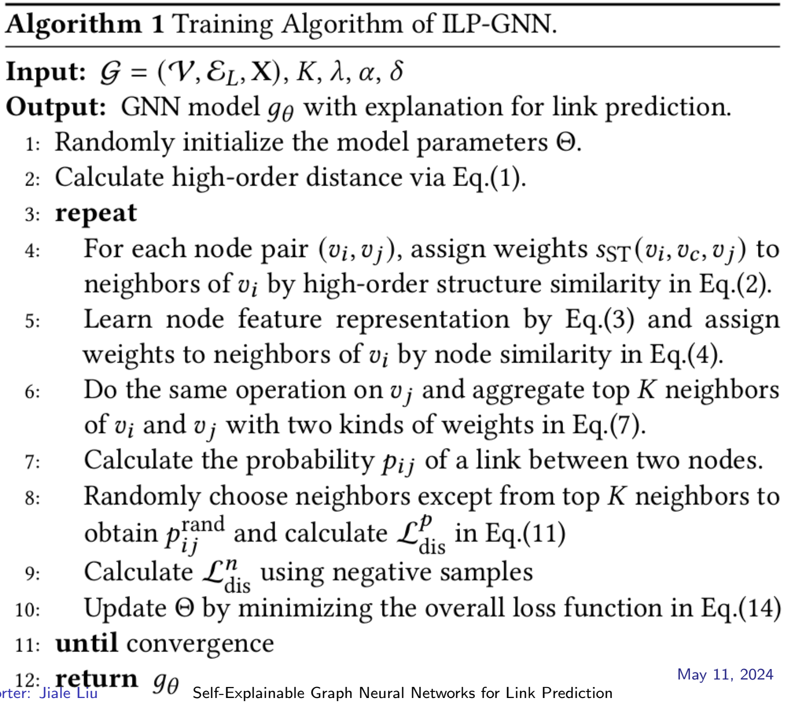
$$\mathcal{L}_{\text{dis}}^{n} = \sum_{e_{ij} \in \mathcal{E}_N} \Bigg(\sum_{v_c \in \mathcal{N}_i^r} s(v_i, v_c, v_j)^2 + \sum_{v_c \in \mathcal{N}_j^r} s(v_j, v_c, v_i)^2 \Bigg). \tag{12}$$

Overall Objective Function

$$\mathcal{L}_{\text{cls}} = \sum_{e_{ij} \in \mathcal{E}_L} -\log p_{ij} + \sum_{e_{ij} \in \mathcal{E}_N} -\log\left(1 - p_{ij}\right), \tag{13}$$

$$\min_{\Theta} \mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda(\mathcal{L}_{\text{dis}}^{p} + \mathcal{L}_{\text{dis}}^{n}), \tag{14}$$


GNN Encoder
Graph Diffusion
Local graph of node pairs
v_c
v_i
v_j
h_j^r
h_c^r
$s_{NO}(v_i, v_c, v_j)$
Node Similarity
Overall Similarity
$s(v_i, v_c, v_j)$
S_{cj}
$s_{ST}(v_i, v_c, v_j)$
High-order Structure Similarity
Explanation Neighbors
Neighbors Aggregation
Interpretable Neighbors Aggregation for v_i
Interpretable Neighbors Aggregation
h_i
h_j
Explanation Neighbors Set
$\mathcal{N}_i^r$
$\mathcal{N}_j^r$
p_{ij}
$\mathcal{L}_{cls}$
e_{ij}
$\mathcal{L}_{dis}^p$
$\mathcal{L}_{dis}^n$
Explanation Enhancement
Overall Framework



Algorithm 1 Training Algorithm of ILP-GNN.

```
Input: $\mathcal{G} = (\mathcal{V}, \mathcal{E}_L, \mathbf{X})$, $K$, $\lambda$, $\alpha$, $\delta$
Output: GNN model $g_\theta$ with explanation for link prediction.
 1: Randomly initialize the model parameters $\Theta$.
 2: Calculate high-order distance via Eq.(1).
 3: repeat
 4:     For each node pair $(v_i, v_j)$, assign weights $s_{\mathrm{ST}}(v_i, v_c, v_j)$ to
        neighbors of $v_i$ by high-order structure similarity in Eq.(2).
 5:     Learn node feature representation by Eq.(3) and assign
        weights to neighbors of $v_i$ by node similarity in Eq.(4).
 6:     Do the same operation on $v_j$ and aggregate top $K$ neighbors
        of $v_i$ and $v_j$ with two kinds of weights in Eq.(7).
 7:     Calculate the probability $p_{ij}$ of a link between two nodes.
 8:     Randomly choose neighbors except from top $K$ neighbors to
        obtain $p_{ij}^{\mathrm{rand}}$ and calculate $\mathcal{L}_{\mathrm{dis}}^{p}$ in Eq.(11)
 9:     Calculate $\mathcal{L}_{\mathrm{dis}}^{n}$ using negative samples
10:     Update $\Theta$ by minimizing the overall loss function in Eq.(14)
11: until convergence
12: return $g_\theta$
```

Catalogs

Introduction

Problems to Solve

Methodology

Prediction

Loss Function

Refrence

[1] Huaisheng Zhu et al. *Self-Explainable Graph Neural Networks for Link Prediction*. 2023. arXiv: 2305.12578 [cs.LG].