

图神经网络理论与应用

图结构学习

授课教师：周晟

浙江大学 软件学院

2024.12



课程内容

- ① 研究背景
- ② 基于优化的图结构学习
- ③ 基于度量学习的图结构学习
- ④ 基于生成模型的图结构学习
- ⑤ GSL 的最新研究进展

① 研究背景

② 基于优化的图结构学习

③ 基于度量学习的图结构学习

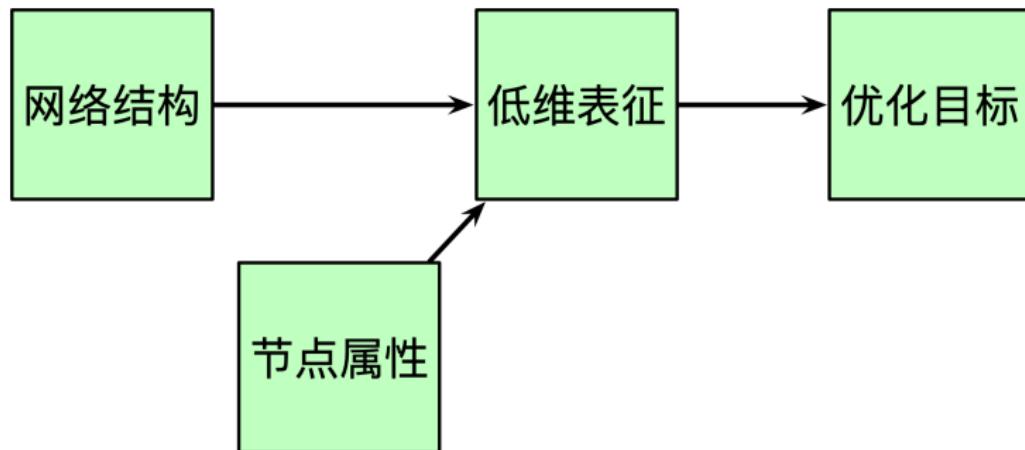
④ 基于生成模型的图结构学习

⑤ GSL 的最新研究进展

研究背景

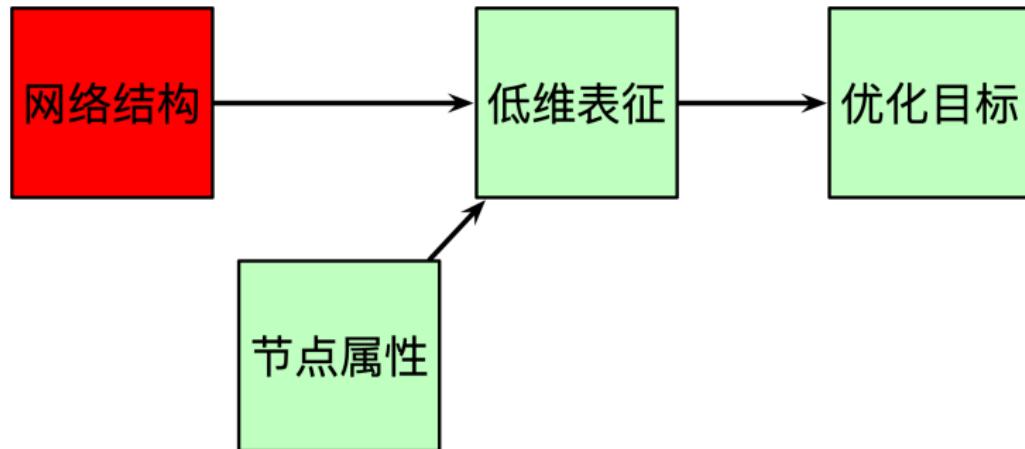
图神经网络的原理

通过将节点的信息沿着网络结构传递（Message Passing），捕获节点间的关系和节点的特征，学习更好的图（节点，边，图）表征。



图神经网络基本框架

研究背景



网络结构受损下的图神经网络

问题与挑战

实际场景中，网络结构往往面临受损或缺失的问题，使得依赖网络结构信息的图神经网络效果严重下降。

图结构缺失问题

- ① 数据天然的稀疏性（社交网络关注量有限）
- ② 数据采集、存储过程中丢失
- ③ 人为隐藏关系（犯罪）
- ④ 数据集本身没有关系型结构



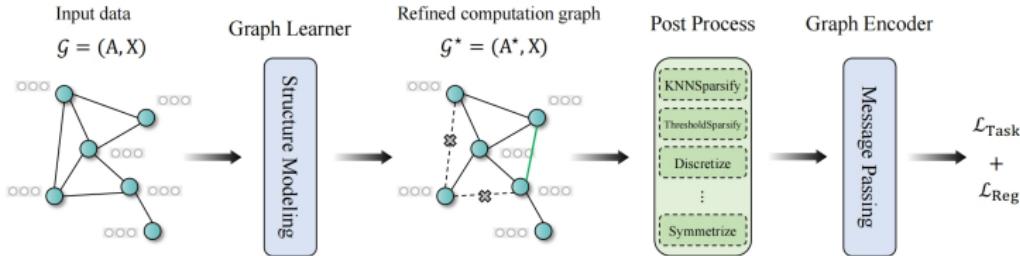
图结构冗余问题

- ① 数据采集、存储过程中错误
- ② 电商网络中，刷单行为（诈骗高发）
- ③ 人为注入，“浑水摸鱼”



图结构学习概念

面对初始图结构存在缺陷的问题，**图结构学习**（Graph Structure Learning, GSL）以一个直接的思想解决——改良原始图结构，学出来一个相对更好的结构。



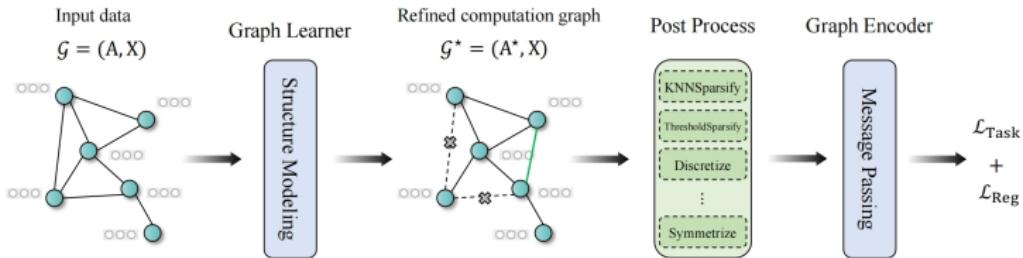
核心在于除了在下游任务上的**表征学习**外，还要学习一个**最优图结构**。

图结构学习框架

基本框架

基于原始图 $G = (\mathbf{A}, \mathbf{X})$ ，我们会学习一个最优图 $\mathbf{A}^* = g_\theta(\mathbf{A}, \mathbf{X})$ ，其中 g_θ 为图结构学习模块，随后将这个最优图输入到 GNN 模块 f_ω ，进行预测，总体目标为：

$$\min_{\omega, \theta} \mathcal{L}_{\text{Task}}(f_\omega(g_\theta(\mathbf{A}, \mathbf{X}), \mathbf{X}), \mathbf{Y}) + \mathcal{L}_{\text{Reg}} \quad (1)$$



图结构学习的意义

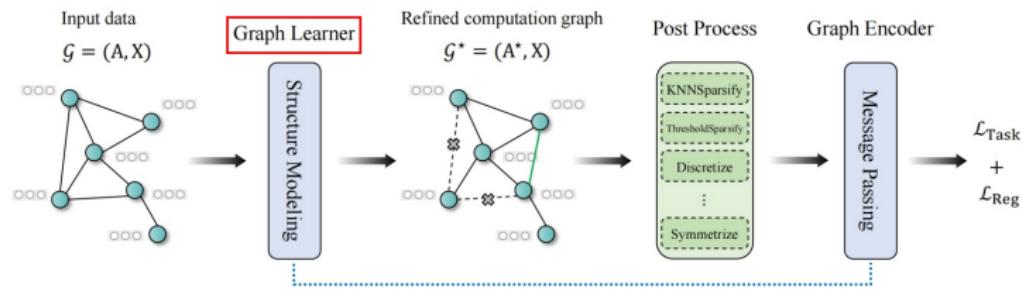
图结构学习对于图神经网络的理论和应用均有重要意义：

- ① 理解数据间的真实关系
- ② 预测图结构演化趋势
- ③ 发掘数据中异常关系

本次课内容：

- ① 图结构学习方法
 - ① 直接优化：将结构作为参数，利用图的性质进行优化
 - ② 度量学习：将网络中的边表示成由图神经网络学习的节点表征的函数
 - ③ 生成模型：假设图结构是生成的，利用生成模型进行优化
- ② 图结构生成方法（Optional）

图结构学习分类



- ① 按照图结构建模方式分类：直接优化方案、基于度量的方法、生成模型方法
- ② 按照优化流程分类：两阶段、端到端、交替

图结构学习分类

按照图结构建模方式分类

- ① 直接优化：将结构作为参数，利用图的性质进行优化
- ② 度量学习：将网络中的边表示成由图神经网络学习的节点表征的函数
- ③ 生成模型：假设图结构是按照某种规律生成的，利用生成模型进行优化

按照优化流程分类

- ① 两阶段方法：先学习结构，再输入 GNN 用于下游任务。
- ② 端到端方法：结构学习模块和 GNN 模块同时优化。
- ③ 交替优化方法：图结构和 GNN 交替更新。

图结构的特性：同配性 (homophily)

同配性 (homophily)

同配性 (homophily)，指一个图网络中相邻节点相似的程度。在节点分类任务中，同配性通常被定义为相邻节点属于同一类别的概率。

可以证明，在同配性为 1 的极端情况下，GNN 可以达到100%的准确率。因此，**如何产生更具有同配性的图结构**是图结构学习中值得思考的问题。

图统计量

除了节点级的局部特征，大规模真实图数据往往具备特殊的宏观性质。

常用如下四种统计量来描述一个图的全局性质：

- ① 度分布，Degree distribution: $P(k)$
- ② 聚类系数，Clustering coefficient: C
- ③ 连通区域，Connected components: s
- ④ 路径长度，Path length: h

图统计量

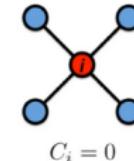
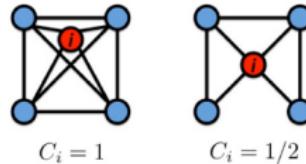
- 度分布, Degree distribution $P(k)$ 是指: 度为 K 的节点数量的分布, $N_k =$ 度为 k 的节点数量,

$$P(k) = N_k/N$$

- 聚类系数, Clustering coefficient

$$C_i = \frac{2e_i}{k_i(k_i - 1)}$$

- e_i 是 i 节点的邻居中互相连边的数量
- k_i 是 i 节点的度

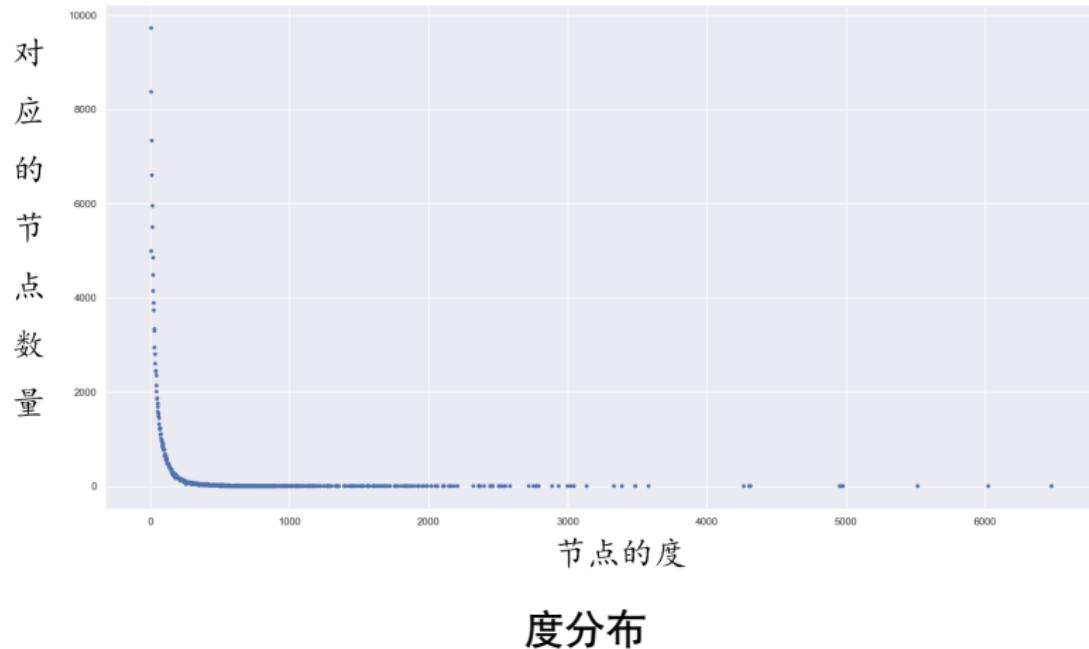


$$C_i \in [0,1]$$

图统计量

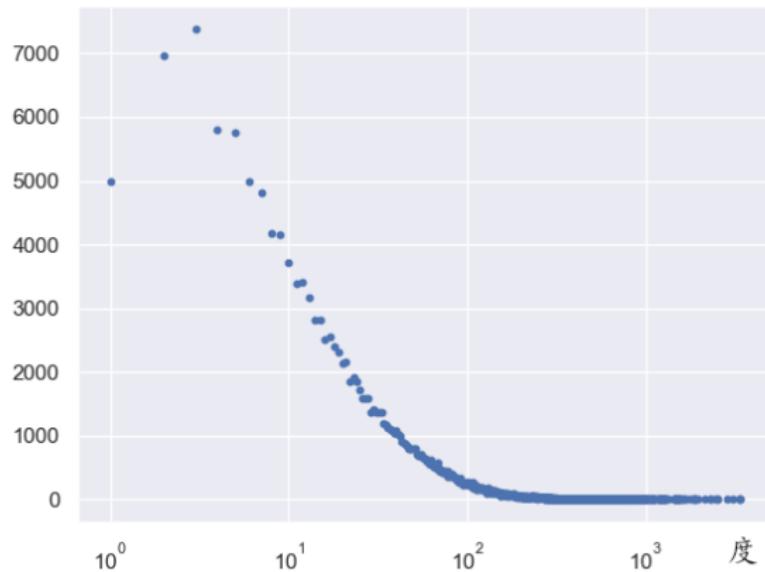
- 连通区域，Connected components，指图中联通区域按大小排序，其不同大小连通区域数量的分布
- 路径长度，Path length，指图中所有存在的最短路径的长度分布
- 以 Twitter 的社交网络图数据集为例，该数据集的邻接矩阵大小为 157575×157575

图统计量-以 Twitter 数据集为例



图统计量-以 Twitter 数据集为例

点的数量



横轴指数化的度分布

随机图的度分布

$$p(k) = \binom{N-1}{K} P^k (1-P)^{N-1-k}$$

- 一个点的度为 k 的概率（有 k 个点与之相连），等价于除它本身之外的 $N-1$ 个点选 k 个和它相连，剩下 $N-1-k$ 和它不连的概率。 P 是两点之间存在一条边的概率， $P = \frac{E}{N*N}$ 。典型二项分布。
- 二项分布可以由泊松分布近似：

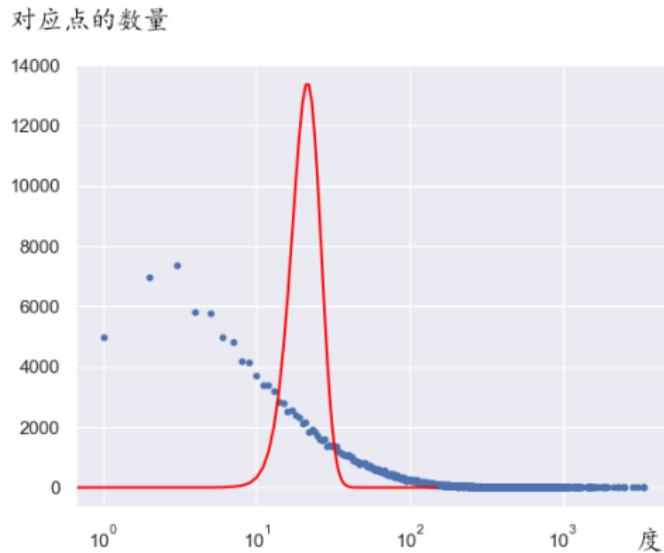
$$p(k) = \binom{N-1}{K} P^k (1-P)^{N-1-k} \approx \frac{\bar{k}^k}{k!} e^{-\bar{k}}$$

其中 $\bar{k} = p(N-1)$ ，是度的均值

- 好的图结构学习结果应保留图的这种天然特性

随机图的度分布

- 以 Twitter 数据集为例，其中共有 157575 点 3530653 边，把真实分布和随机分布（红色）画在一起



① 研究背景

② 基于优化的图结构学习

③ 基于度量学习的图结构学习

④ 基于生成模型的图结构学习

⑤ GSL 的最新研究进展

基于优化的图结构学习

研究动机

图结构可以看作节点之间是否有边的参数的集合，可以直接对图结构进行参数化学习。

关键问题

- ① 优化目标如何确定？
- ② 怎么联合学习图结构和 GNN？

Graph Structure Learning for Robust Graph Neural Networks¹

真实图结构的性质 (properties) :

- ① 低秩
- ② 稀疏
- ③ 邻接的节点特征相近

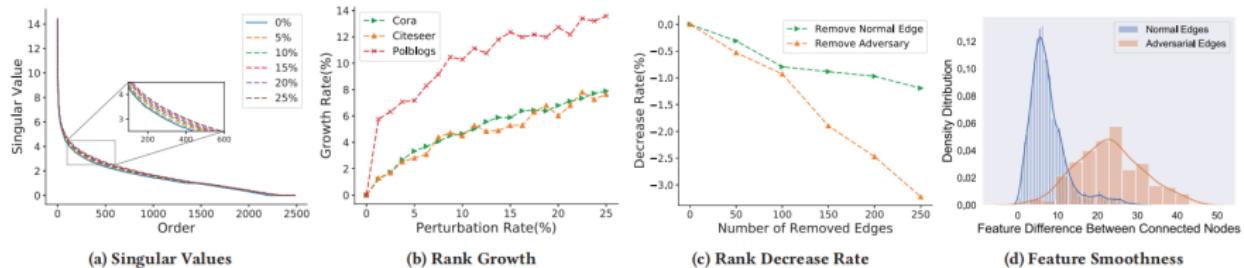


Figure 1: An illustrative example on the property changes of the adjacency matrix by adversarial attacks

¹[Jin et al., 2020]

Pro-GNN: 性质

1. (优化目标) 如何利用这些性质引导图结构的学习?

ProGNN 采取了直接优化邻接矩阵 S 的方法, 将 S 视作一个 $n * n$ 的参数矩阵, 这样这些性质可以表达为有关 S 的一系列损失函数。

图性质的损失函数

① 低秩和稀疏:

$$\mathcal{L}_0 = \|A - S\|_F^2 + \alpha\|S\|_1 + \beta\|S\|_*, s.t., S = S^T$$

② 特征平滑:

$$\mathcal{L}_s = \text{tr}(X^T \hat{L} X) = \frac{1}{2} \sum_{i,j=1}^N S_{ij} \left(\frac{x_i}{\sqrt{d_i}} - \frac{x_j}{\sqrt{d_j}} \right)^2, s.t., S = S^T$$

Pro-GNN: 优化

2. (优化流程) 如何联合学习图结构和图神经网络 ?

最终损失:

$$\begin{aligned} \arg \min_{S \in \mathcal{S}, \theta} \mathcal{L} &= \mathcal{L}_0 + \lambda \mathcal{L}_f + \gamma \mathcal{L}_{GNN} \\ &= \|A - S\|_F^2 + \alpha \|S\|_1 + \beta \|S\|_* + \gamma \mathcal{L}_{GNN}(\theta, S, X, \mathcal{Y}_L) \\ &\quad + \lambda \text{tr}(X^T \hat{L} X), \text{s.t. } S = S^T \end{aligned}$$

由于一起求解较为困难, 使用**交替优化** (Alternative Optimization) 迭代求解。

交替优化

交替优化经常在 GSL 方法中出现, 用来解决双层优化等较困难的优化问题。其做法是固定一个, 优化另一个, 交替进行。

Pro-GNN: 优化

$$\begin{aligned} \arg \min_{S \in \mathcal{S}, \theta} \mathcal{L} = & \|A - S\|_F^2 + \alpha \|S\|_1 + \beta \|S\|_* + \gamma \mathcal{L}_{GNN}(\theta, S, X, \mathcal{Y}_L) \\ & + \lambda \text{tr}(X^T \hat{L} X), \text{s.t. } S = S^T \end{aligned}$$

优化 GNN:

$$\theta \leftarrow \eta' \frac{\partial \mathcal{L}_{GNN}(\theta, S, X, \mathcal{Y}_L)}{\partial \theta}$$

优化图结构 S : 核范数和 l_1 范数不可导, 需使用 Proximal Optimization 数学工具²

²[Jin et al., 2020]

Learning Discrete Structures for Graph Neural Networks³

1. (优化目标) 基于双层优化的思路:

- 外层优化: 在验证集上学习离散结构
- 内层优化: 在训练集上学习 GCN 的参数

$$\min_A F(w_A, A) \text{ such that } w_A \in \arg \min_w L(w, A)$$

$$L(w, A) = \sum_{v \in V_{train}} l(f_w(X, A)_v, y_v) + \Omega(w)$$

$$F(w_A, A) = \sum_{v \in V_{val}} l(f_{w_A}(X, A)_v, y_v)$$

其中内层优化使用训练集中的点, 外层优化使用验证集中的节点。

³[Franceschi et al., 2019]

LDS: Reparameterization

A 是离散变量，无法直接求导，LDS 假设 A 从多个独立的伯努利分布中生成： $A \sim \text{Ber}(\theta)$, $\theta \in (0, 1)^{n \times n}$
优化问题变为：

$$\min_{\theta} \mathbb{E}_{A \sim \text{Ber}(\theta)} [F(w_{\theta}, A)]$$

$$\text{such that } w_{\theta} = \arg \min_w \mathbb{E}_{A \sim \text{Ber}(\theta)} [L(w, A)]$$

LDS: GCN 内层优化

2. (优化流程) 对双层优化问题使用交替优化

内层问题:

$$w_\theta = \arg \min_w \mathbb{E}_{A \sim \text{Ber}(\theta)} [L(w, A)]$$

$$\mathbb{E}_{A \sim \text{Ber}(\theta)} [L(w, A)] = \sum_{A \in \mathcal{H}_N} P_\theta(A) L(w, A)$$

上式不可能直接求出, 按照 SGD 的思想迭代近似, 每次采样一个结构梯度下降:

$$w_{\theta,t+1} = w_{\theta,t} - \gamma \nabla L(w_{\theta,t}, A_t), A_t \sim \text{Ber}(\theta)$$

LDS：图结构外层优化

外层问题：

$$\min_{\theta} \mathbb{E}_{A \sim \text{Ber}(\theta)} [F(w_{\theta}, A)]$$

求解对 θ 的梯度，使用链式法则分解：

$$\begin{aligned}\nabla_{\theta} \mathbb{E}_{A \sim \text{Ber}(\theta)} [F(w_{\theta,T}, A)] &\approx \mathbb{E}_{A \sim \text{Ber}(\theta)} [\nabla_A F(w_{\theta,T}, A)] \\ &= \mathbb{E}_A [\partial_w F(w_{\theta,T}, A) \nabla_A w_{\theta,T} + \partial_A F(w_{\theta,T}, A)]\end{aligned}$$

同样使用 SGD 迭代近似最优解：

$$\theta' = \theta - \eta (\partial_w F(w_{\theta,T}, A) \nabla_A w_{\theta,T} + \partial_A F(w_{\theta,T}, A)), A \sim \text{Ber}(\theta)$$

LDS: 交替优化

进行 T 次内层迭代：

$$w_{\theta,t+1} = w_{\theta,t} - \gamma \nabla L(w_{\theta,t}, A_t), A_t \sim \text{Ber}(\theta)$$

进行一次外层迭代：

$$\theta' = \theta - \eta (\partial_w F(w_{\theta,T}, A) \nabla_A w_{\theta,T} + \partial_A F(w_{\theta,T}, A)), A \sim \text{Ber}(\theta)$$

其中外层梯度计算中第一项 $\nabla_A w_{\theta,T}$ 可在内层迭代中回溯
(涉及超梯度的计算方式, 使用 Straight-through Estimator, STE)。

小结

直接优化方案将邻接矩阵设置为 $O(n^2)$ 的参数矩阵，简单直观，但计算代价很大，不适用于大规模图。

1

研究背景

2

基于优化的图结构学习

3

基于度量学习的图结构学习

4

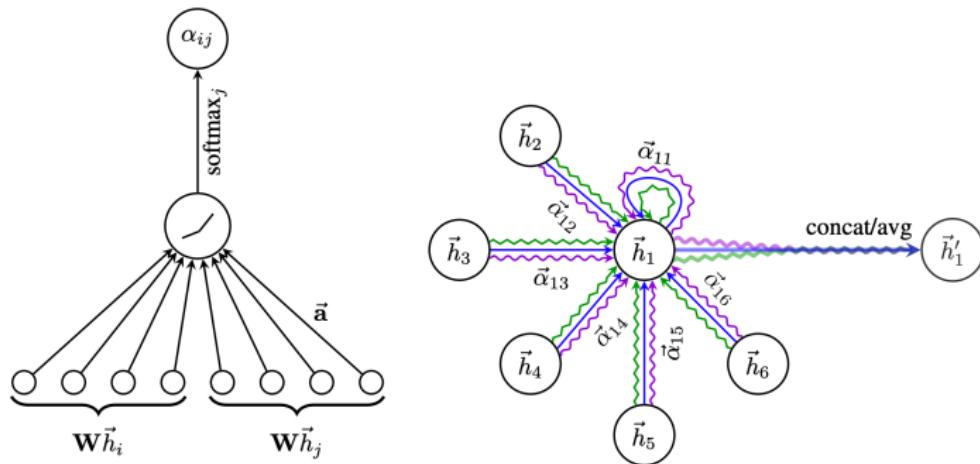
基于生成模型的图结构学习

5

GSL 的最新研究进展

GAT 与图结构学习

$$\alpha_{ij} = \frac{\exp(\text{LeakReLU}(\vec{a}^T [W\vec{h}_i || W\vec{h}_j]))}{\sum_{k \in \mathcal{N}_i} \exp(\text{LeakyReLU}(\vec{a}^T [W\vec{h}_i || W\vec{h}_k]))}$$

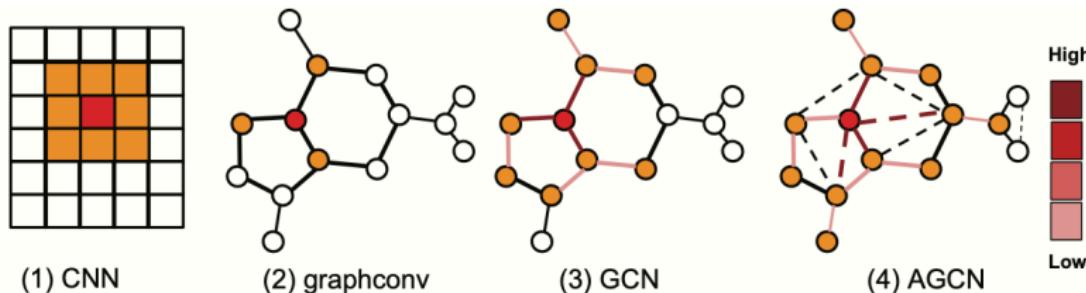


GAT 可看作一种特殊的图结构学习方法。

Adaptive Graph Convolutional Neural Networks(AGCN)⁴

研究动机

现有的图神经网络采用消息传递机制，但仅能在已观测的边上传递信息。由于网络稀疏性问题，实际可传递有效信息的节点并没有被连接。



AGCN 与 CNN、图卷积和 GCN 的对比

⁴[Li et al., 2018]

GCN 模型更新方式：

$$H^{(l+1)} = \sigma \left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)} \right)$$

一个新的图结构即对应一个新的**拉普拉斯矩阵**。

AGCN 对图结构学习的设想

- 图结构可参数化学习
- 参数量少（拉普拉斯矩阵参数化需要 $O(N^2)$ 的空间）
- 可被应用于不同的图结构（可优化更多任务，如点云分类）

解决方案：基于核（kernel）的方法

AGCN 的图结构学习策略

使用广义 Mahalanobis 距离实现基于核的距离度量

$$\mathbb{D}(x_i, x_j) = \sqrt{(x_i - x_j)^T M (x_i - x_j)}$$

- x_i 是图卷积网络某一层中节点 i 的特征。
- $M = W_d W_d^T$, $W_d \in \mathbb{R}^{d \times d}$ 是可学习的 kernel。

将参数量从 $O(N^2)$ 降至 $O(d^2)$!

Adaptive Graph Convolutional Neural Networks

给定距离度量，使用 Gaussian Kernel 归一化得到稠密的邻接矩阵 \tilde{A} 。

$$\tilde{A}_{ij} \leftrightarrow \mathbb{G}_{x_i, x_j} = \exp(-\mathbb{D}(x_i, x_j)/(2\sigma^2))$$

将其作为残差结构与原图结构叠加，得到更新后的拉普拉斯矩阵并用于卷积图神经网络：

$$\hat{L} = L + \alpha L_{res} = L + \alpha(I - \tilde{D}^{-1/2} \tilde{A} \tilde{D}^{-1/2})$$

为什么要加上原来的图结构？

总结与归纳：

优点

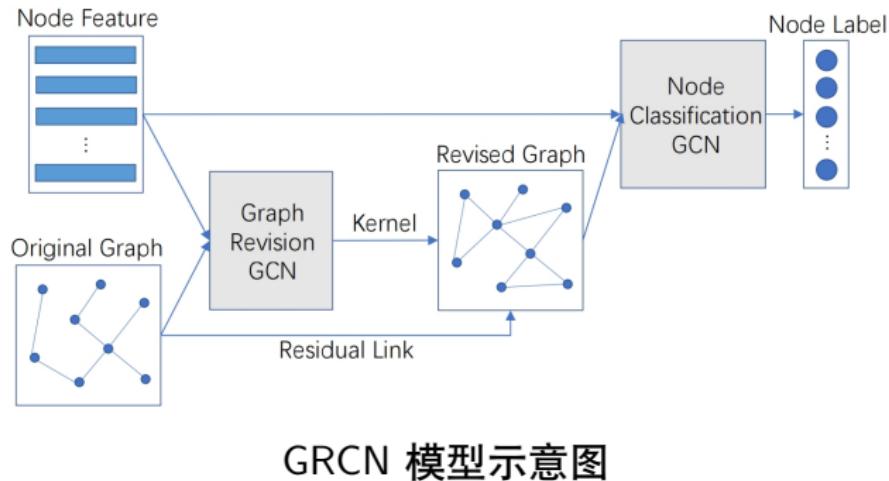
- ① 模型简单，启发了很多后续的图结构学习工作
- ② 参数量小，便于优化

缺点

- ① 结构仅通过线性变换学习
- ② 学习的拉普拉斯矩阵过于稠密
- ③ 图神经网络没有给结构学习带来信息
- ④ 模型缺乏明确的结构学习优化目标

Graph-Revised Convolutional Network⁵

GRCN 使用经过 GNN 编码的节点表征来计算图结构，可以视作在 GNN 前拼接了一个 Link Prediction 模块

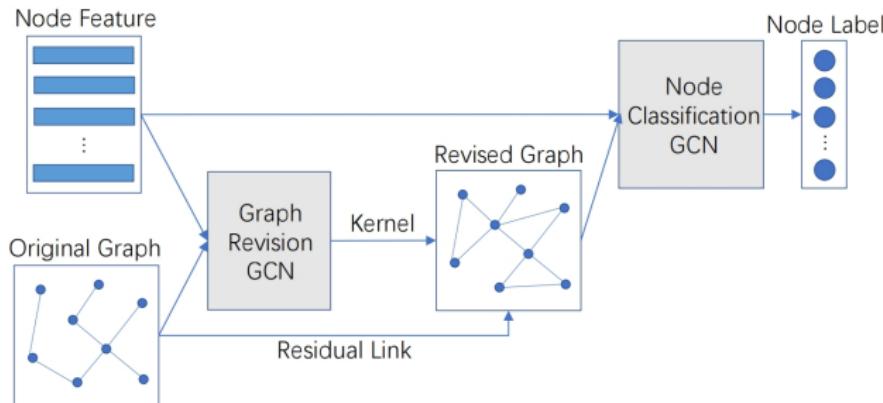


⁵[Yu et al., 2021]

Graph-Revised Convolutional Network

计算流程

- ① 使用 GNN Encoder 得到节点表征 $\mathbf{F} = \text{GNN}_s(\mathbf{A}, \mathbf{X})$
- ② 计算 pairwise similarity $\hat{\mathbf{A}}_{ij} = \cos(\mathbf{F}_i, \mathbf{F}_j)$
- ③ 稀疏化、对称化、残差连接后处理操作
 $\mathbf{A}^* = \mathbf{A} + \text{sym} \circ \text{topk}(\hat{\mathbf{A}})$
- ④ 输入下游任务的 $\text{GNNY}_{\text{pred}} = \text{GNN}(\mathbf{A}^*, \mathbf{X})$



GRCN 模型示意图

Learning to Drop: Robust Graph Neural Network via Topological Denoising⁶

研究动机

网络中观察到的边往往存在与任务无关的边，通过这些节点传递信息往往给节点表征学习带来噪声。以节点分类任务为例，标签一致的节点才能提供有效的分类信息。

		Ratio of positive edges removed										
		0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Ratio of negative edges removed	0	60.1	60.1	55.7	55.2	54.8	54.8	54.2	53.8	53.6	53.5	53.5
	0.1	64.2	63.7	61.4	60.0	59.6	59.4	59.3	58.7	58.7	58.6	57.4
	0.2	69.6	68.2	66.5	66.4	66.1	65.4	63.6	63.8	62.6	62.1	61.2
	0.3	72.8	72.3	71.5	70.5	70.2	69.0	68.3	67.7	68.9	67.6	66.8
	0.4	79.3	76.9	74.5	73.5	73.5	72.9	72.6	71.8	71.2	70.3	69.5
	0.5	80.4	79.2	78.0	76.6	75.6	75.3	75.1	74.3	73.7	73.6	72.3
	0.6	83.6	82.4	81.3	80.6	80.3	78.6	78.1	77.3	76.8	75.0	74.1
	0.7	83.9	82.6	81.6	81.5	81.0	80.1	79.5	78.2	78.1	77.7	76.5
	0.8	85.5	83.8	83.5	82.8	81.1	80.7	80.7	79.9	79.6	79.9	79.4
	0.9	86.3	86.1	84.8	83.6	83.6	82.6	82.4	81.8	81.3	81.1	81.0
	1	87.2	86.2	85.3	85.1	84.3	84.1	84.0	83.0	82.1	82.1	81.1

PTDNet: 图的稀疏化

图的稀疏化，即将 dropout 的思想用于图结构以加强模型鲁棒性，从图中删除一部分边。

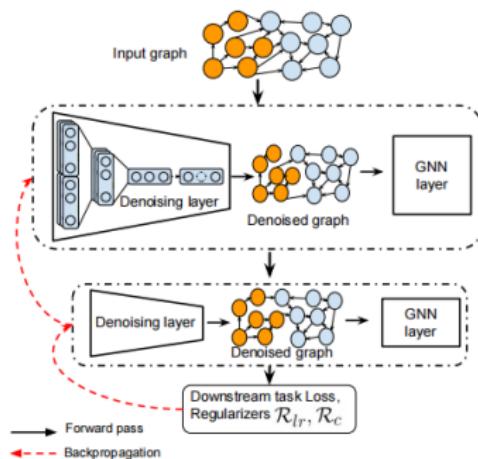
具体可分为：随机去边、根据预先定义的规则、参数化三种。

图的稀疏化和图结构学习

- ① 图的稀疏化可以视作一种图结构学习，但仅在初始存在的边上进行，即可以去除冗余的边，但不能增加新边（后者往往需要耗费更多空间）。
- ② 该不同之处通常表现在得到新的结构后与原结构乘还是加。

PTDNet

PTDNet (parameterized topological denoising network), **参数化图结构降噪网络**, 使这一过程受任务损失监督, 并联合优化 GNN 网络和降噪网络。

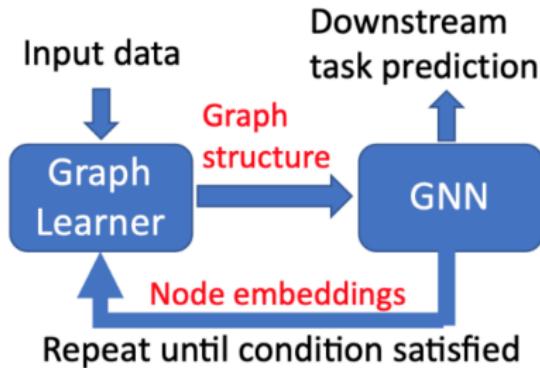


每层 GCN 均配备一个降噪网络, 使用上一层表征产生新结构给该卷积层使用。(整体架构和 GAT 类似)

Iterative Deep Graph Learning for Graph Neural Networks: Better and Robust Node Embeddings⁷

研究动机

图结构与图神经网络学习可以相互帮助：好的图结构可以学习好的图神经网络，好的图神经网络结果可以帮助建模更准确的网络结构。



⁷[Chen et al., 2020]

从图结构学习到图神经网络

- 图结构沿用 AGCN 的思想：融合原始网络结构与学习过程中的网络结构

$$\tilde{\mathbf{A}}^{(t)} = \lambda \mathbf{L}^{(0)} + (1 - \lambda) \{ \eta \mathbf{f}(\mathbf{A}^{(t)}) + (1 - \eta) \mathbf{f}(\mathbf{A}^{(1)}) \}$$

- 图神经网络沿用经典的 GCN：

$$\mathbf{Z} = \text{ReLU} \left(\mathbf{MP}(\mathbf{X}, \tilde{\mathbf{A}}) \mathbf{W}_1 \right), \hat{\mathbf{y}} = \sigma \left(\mathbf{MP}(\mathbf{Z}, \tilde{\mathbf{A}}) \mathbf{W}_2 \right)$$

$$\mathcal{L}_{\text{pred}} = \ell(\hat{\mathbf{y}}, \mathbf{y})$$

根据图神经网络学习图结构

基本假设

具有相似的低维表征的节点对，有更大的概率生成边。

基于节点表征的边概率计算：

- 第一层：基于向量的相似度计算

$$S_{ij} = \cos(\vec{w}_p \odot \vec{z}_i, \vec{w}_p \odot \vec{z}_j)$$

- 第二层：基于多头向量的相似度计算

$$S_{ij} = \frac{1}{m} \sum_{p=1}^m \cos(\vec{w}_p \odot \vec{z}_i, \vec{w}_p \odot \vec{z}_j)$$

根据图神经网络学习图结构

使用节点表征相似度预测图结构的缺陷：

- ① 计算复杂度高
- ② 网络过于稠密

IDGL 的解决方案：

- 第三层：Anchor-based 度量学习：

$$a_{ik}^p = \cos(\mathbf{w}_p \odot \mathbf{v}_i, \mathbf{w}_p \odot \mathbf{u}_k), \quad a_{ik} = \frac{1}{m} \sum_{p=1}^m a_{ik}^p$$

- 第四层：图稀疏化。只选择每个节点最近的 K 个节点，或相似度大于阈值 ϵ 的邻居节点。

图结构质量控制

IDGL 迭代优化的缺陷

图结构学习模块缺少监督信号，容易陷入局部平凡解 (Trival Solution)

好的图结构应具有如下性质：

- ① 平滑性：临近的节点的属性相似度高

$$\Omega(\mathbf{A}, \mathbf{X}) = \frac{1}{2n^2} \sum_{i,j} A_{ij} \|\mathbf{x}_i - \mathbf{x}_j\|^2 = \frac{1}{n^2} \text{tr} (\mathbf{X}^T \mathbf{L} \mathbf{X})$$

- ② 连通性：网络中的节点尽可能构成联通图
- ③ 稀疏性：网络中的边总体稀疏

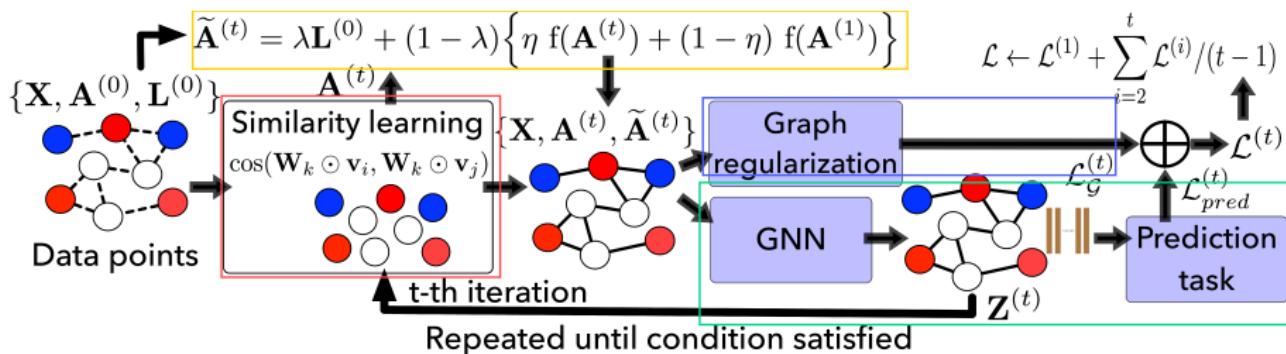
$$f(\mathbf{A}) = \frac{-\beta}{n} \mathbf{1}^T \log(\mathbf{A} \mathbf{1}) + \frac{\gamma}{n^2} \|\mathbf{A}\|_F^2$$

IDGL 模型

在 t 时刻：

- ① 基于当前节点表征计算边权值
- ② 融合原始结构
- ③ 使用 GNN 计算点表征，进行预测
- ④ 同时计算对结构的约束项

迭代计算直至收敛



IDGL 方法示意图

循环优化图神经网络

一般形式：

- ① 使用已有结构学习节点表征 z

$$\mathbf{z} = GNN(X, A)$$

- ② 利用节点表征相似度学习图结构：

$$\tilde{\mathbf{A}}_{ij} = \phi(\mathbf{z}_i, \mathbf{z}_j)$$

- ③ 融合输入结构和学习图结构

$$\mathbf{A}^* = g(\mathbf{A}, \tilde{\mathbf{A}})$$

迭代循环，直至收敛。

基于节点相似度的图结构生成

① Gaussian Kernel

$$\begin{aligned}\phi(\mathbf{z}_i, \mathbf{z}_j) &= \sqrt{(\mathbf{z}_i - \mathbf{z}_j)^\top \mathbf{M} (\mathbf{z}_i - \mathbf{z}_j)} \\ \tilde{\mathbf{A}}_{ij} &= \exp\left(-\frac{\phi(\mathbf{z}_i, \mathbf{z}_j)}{2\sigma^2}\right)\end{aligned}$$

② 向量内积

$$\tilde{\mathbf{A}} = \sigma(\mathbf{Z} \mathbf{Z}^\top)$$

③ Cosine 相似度

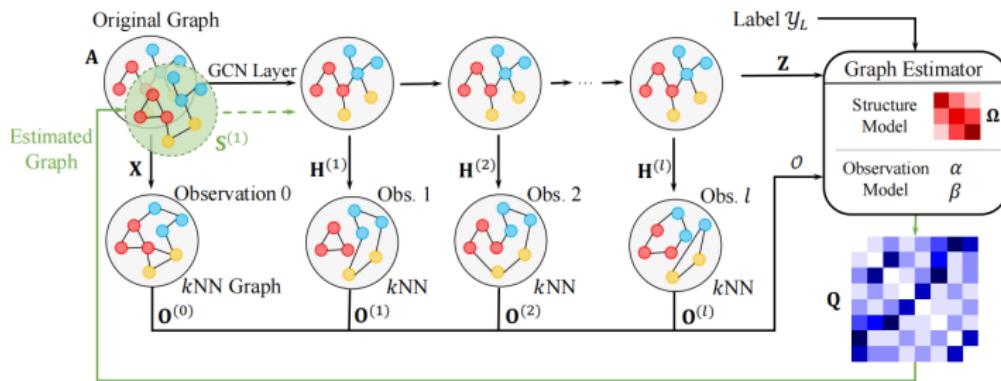
$$\phi(\mathbf{z}_i, \mathbf{z}_j) = \frac{\mathbf{z}_i \mathbf{z}_j^\top}{\|\mathbf{z}_i\|_2 \|\mathbf{z}_j\|_2}.$$

- 1 研究背景
- 2 基于优化的图结构学习
- 3 基于度量学习的图结构学习
- 4 基于生成模型的图结构学习
- 5 GSL 的最新研究进展

Graph Structure Estimation Neural Networks⁸

研究动机

现有的图结构学习方法仅考虑图数据的判别特性（节点之间是否有边），对网络的生成过程缺乏可解释性。



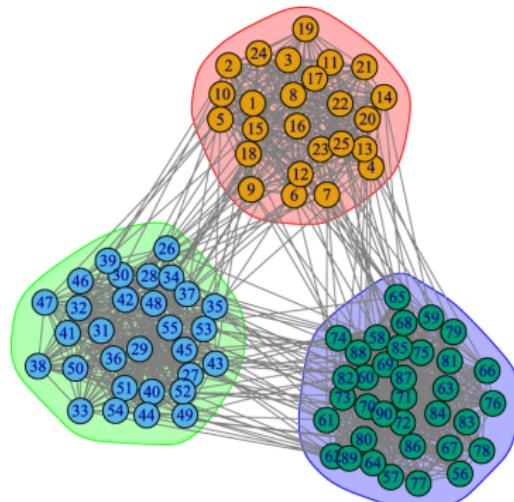
GEN 的模型示意图

⁸[Wang et al., 2021]

SBM 模型简介

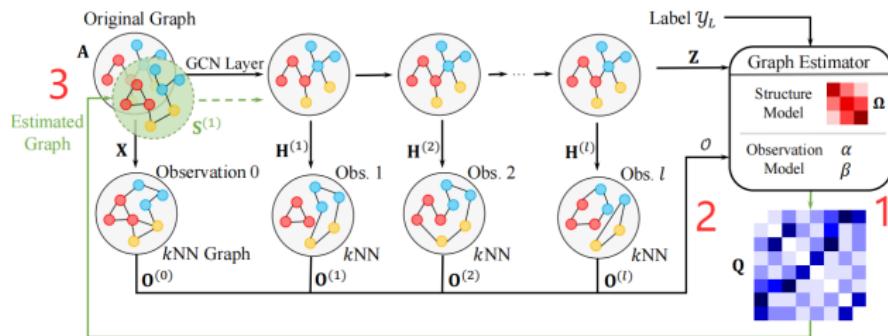
Stochastic Block Model(SBM)

Stochastic Block Model(SBM) 是一种经典的网络生成模型。它假设网络中边的生成是由两个节点的社区分布决定，即属于同一个社区的节点有更大的概率生成边，不同社区的节点生成边的概率更小。



GEN 的基本思路：

- ① 假设存在一个图的**最优结构**，从一个**SBM 模型**中生成。
- ② 假设该最优结构存在多个**观测**，从**节点不同阶段的表示**中各**自生成 KNN 图**作为观测。
- ③ **交替优化**，在训练好的 GCN 基础上使用**EM 算法**求解最优结构，将其作为新的结构重新训练。



GEN 的模型示意图

GEN 的图结构建模

模型假设

最优图结构 A^* 由一个隐藏的 SBM 模型产生。

在 SBM 模型的假设下，网络中存在边的概率只与两个节点所属的社区有关。给定参数矩阵 $\Omega \in \mathcal{S}^{|C|*|C|}$ ，网络的生成概率为：

$$P(A^* | \Omega, Z, \mathcal{Y}_L) = \prod_{i < j} \Omega_{c_i c_j}^{A_{ij}^*} (1 - \Omega_{c_i c_j})^{1 - A_{ij}^*}$$

c_i 表示节点 i 的类别。当其属于训练集时，使用真实标签；当属于无标签节点时，使用预测标签。

$$c_i = \begin{cases} y_i & \text{if } v_i \in \mathcal{V}_L \\ z_i & \text{otherwise} \end{cases}$$

GEN 的图结构建模

模型假设

给定最优的图结构 A^* , 真实观测的图结构由最优图结构随机采样后生成。

定义最优图结构下有边和没有边, 在观测图结构下存在边的概率为 α , β , 观测图结构的概率为:

$$P(O | A^*, \alpha, \beta) = \prod_{i < j} [\alpha^{E_{ij}} (1 - \alpha)^{M - E_{ij}}]^{A_{ij}^*} \times [\beta^{E_{ij}} (1 - \beta)^{M - E_{ij}}]^{1 - A_{ij}^*}$$

其中 E_{ij} 为给定的一组 M 个观测图结构中 ij 之间出现边的次数



GEN 的整体建模

GEN 的模型优化

GEN 模型需要优化的参数有最优图结构 A^* 和图神经网络参数 ω 。
对于高度耦合的两种参数，使用交替优化：

优化 GNN:

$$\omega \leftarrow \eta' \frac{\partial \mathcal{L}_{GNN}(\omega, S, X, \mathcal{Y}_L)}{\partial \omega}$$

利用原图结构和属性图作为输入优化图结构 S :

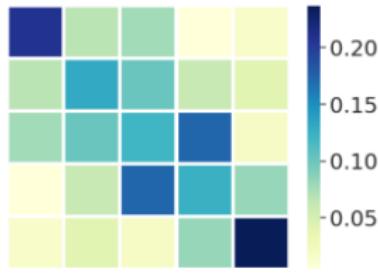
$$A^* \leftarrow EM(O^{(0)}, O^{(1)}, O^{(2)}, \dots, O^{(l)}, O^{(l+1)})$$

式中 $O^{(i)}$ 为不同阶段的表征产生的 KNN 图。

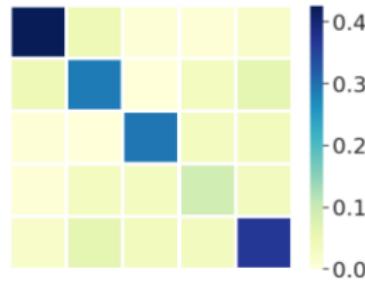
GEN: 同配性

在 toy dataset 上验证了: GEN 可以增加相同类别节点之间的边, 减少连接不同类别的边。

体现了 GEN 可以学习具有同配性的图结构的能力。



(a) Original graph



(b) Estimated graph

GEN 产生结构的同配性

- ① 研究背景
- ② 基于优化的图结构学习
- ③ 基于度量学习的图结构学习
- ④ 基于生成模型的图结构学习
- ⑤ GSL 的最新研究进展
 - 无监督 GSL
 - 层次化 GSL
 - GSL 在其他领域的应用
 - GSL Benchmark

GSL 的最新研究

GSL 的最新研究关注图建模方式以外的更多方面，这些工作包括：

- ① 无监督的图结构学习：SUBLIME(WWW 22), STABLE(WWW 22), GSR(WSDM 23)
- ② 其他改进：PROSE(层次化学习, KDD 23), GraphGLOW(KDD 23)
- ③ 图结构学习在其他任务上的应用：VIB-GSL(图分类, AAAI 22), HoLe(聚类, CIKM 23), RDGSL(动态图, CIKM23), DCGC (信心校准, KDD24), DSLR (连续学习, WWW24)
- ④ Benchmark: OpenGSL(NIPS 23)

Unsupervised GSL

以上的方法都基于有监督情况，结构学习依赖监督信息

Drawbacks

- ① 不平衡。标签有限时有标签节点周围边受到监督信息引导，但距离有标签节点距离远的边难以学习。
- ② 不实用。相较于监督情况，无监督情况更广泛。
- ③ 不通用。对于每个任务都要重新学习结构，不能学习一个通用的结构。

近期出现的 Unsupervised GSL 工作大多利用了对比学习的思想，关键在于如何设计不同的 view，如何融入结构学习。

Reliable Representations Make A Stronger Defender: Unsupervised Structure Refinement for Robust GNN_[Li et al., 2022]

研究动机

聚焦基于距离度量的 GSL 方法，认为监督情况下得到的表征容易受扰动，本身质量不佳，不能产生好的结构。

提出了一个 two-stage 的无监督方法 STABLE，分为两个阶段。

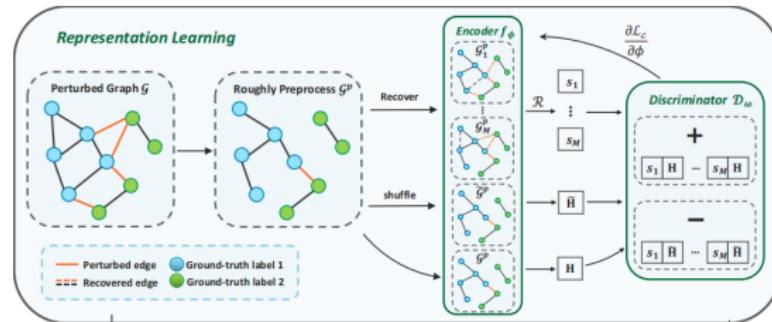
Representation Learning：用**对比学习**得到 reliable 的表征

Graph Refining：根据该表征的相似度计算结构。

Representation Learning

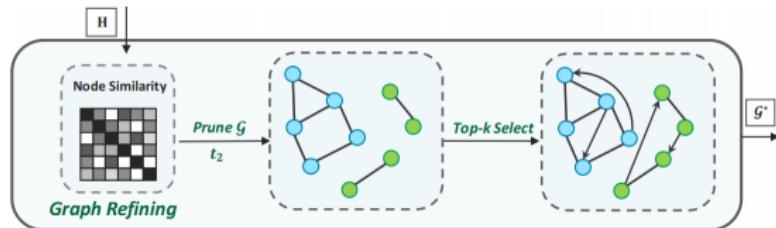
如何设计不同的 view，使得学到的表征可以对抗噪声？
对于一个含噪声的图 G ：

- ① 采取一个简单预处理，删除一部分边，得到噪声程度低的图 G^p
- ② 从 G^p 中随机恢复被删除的边，构成 M 个噪声程度较高的图 $G_1^p, G_2^p, \dots, G_M^p$
- ③ 要求从 G^p 和 $G_1^p, G_2^p, \dots, G_M^p$ 得到的表征接近，同时 shuffle G^p 得到 \tilde{G}^p 作为负样本，要求 \tilde{G}^p 和 $G_1^p, G_2^p, \dots, G_M^p$ 得到的表征远离



Graph Refining

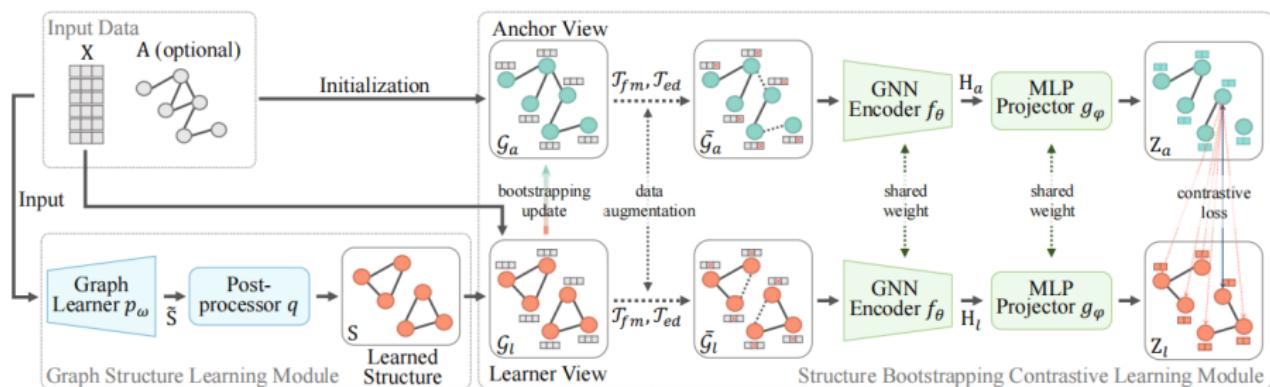
在获得 reliable 的表征 H 后，使用常规的相似度计算和后处理得到最终的结构：



该结构与任务无关的，从无监督方法获得，可以直接用于多个下游任务。

Towards Unsupervised Deep Graph Structure Learning⁹

可学习的 GSL 模块产生 learner view，原结构作为 anchor view，最大化两个 view 之间的互信息 (MI)。



⁹[Liu et al., 2022]

涵盖了之前 GSL 工作中的多种产生图的方法：

1. 直接将整个图作为参数：

$$S = g(\Omega), \Omega \in R^{n \times n}$$

2. Learnable encoder+metric：

$$S = metric(encoder_w(X, A))$$

再经过后处理操作（对称化、稀疏化等）得到 learner view

$$G_l = (S, X)$$

Contrastive Learning

得到 learner view $G_l(A, S)$ 后和预先指定的 anchor view (文中设置为原图 $G_a = (A, X)$) 进行对比学习。

使用共享的 GCN+MLP 得到表征：

$$Z_l = g_\phi(f_\theta(\overline{G}_l)), Z_a = g_\phi(f_\theta(\overline{G}_a))$$

对比学习最大化互信息：

$$\mathcal{L} = \frac{1}{2n} \sum_{i=1}^n [\ell(z_{l,i}, z_{a,i}) + \ell(z_{a,i}, z_{l,i})]$$
$$\ell(z_{l,i}, z_{a,i}) = \log \frac{e^{\text{sim}(z_{l,i}, z_{a,i})/t}}{\sum_{k=1}^n e^{\text{sim}(z_{l,i}, z_{a,k})/t}}$$

更新 Anchor view

一直使用原图作为 anchor view 会带来原结构中的误差，文中每隔若干轮次更新，和当前学习的结构加权：

$$A_a = \tau A_a + (1 - \tau)S$$

PROSE: Graph Structure Learning via Progressive Strategy¹⁰

研究动机

现有的图结构学习方法同时学习图结构的不同部分，但一张图中各部分**重要性不同**，应当先学习**重要节点之间的连接**。如社交网络中先学习名人之间的连接。

困难与挑战

- ① 如何得到图结构中不同区域的重要性排序？
- ② 如何根据重要性依次学习结构？

¹⁰[Wang et al., 2023]

PROSE: 问题形式化

$$P_{\phi}(A^* | G) \propto \sum_{(A_m, \dots, A_1) \sim \prod_{i=1}^m P_{\phi_i^{pss}}(A_i | A_{>i}, \mathcal{G})} \prod_{i=1}^m P_{\phi_i^{psr}}(A_i^* | A_{<i}^*, A_i)$$

PROSE 基于应当按照重要性次序学习不同子结构的思想，将问题形式化为两个模块。

- Progressive Structure Splitting(PSS):

$$(A_m, \dots, A_1) \sim \prod_{i=1}^m P_{\phi_i^{pss}}(A_i | A_{>i}, \mathcal{G})$$

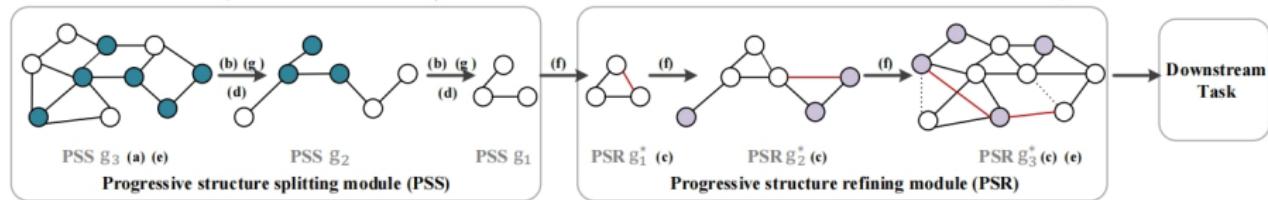
自顶向下划分原结构为重要性由小到大的层级

- Progressive Structure Refining(PSR):

$$\prod_{i=1}^m P_{\phi_i^{psr}}(A_i^* | A_{<i}^*, A_i)$$

自底向上依次学习结构，重要的子结构优先学习。

PROSE: 模型



PROSE 方法示意图

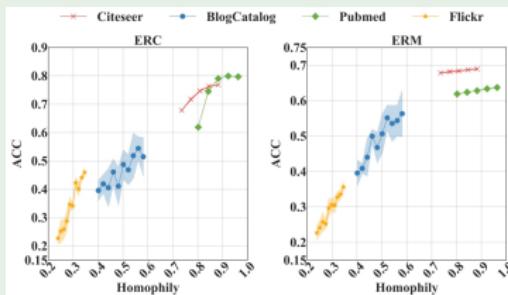
- 按照渐进策略，使用 Graph Pooling 划分原始结构，再自底向上学习结构
- 事实上形成了一种 encoder-decoder 架构

Homophily-enhanced Structure Learning for Graph Clustering¹¹

GSL+Graph Clustering

研究动机

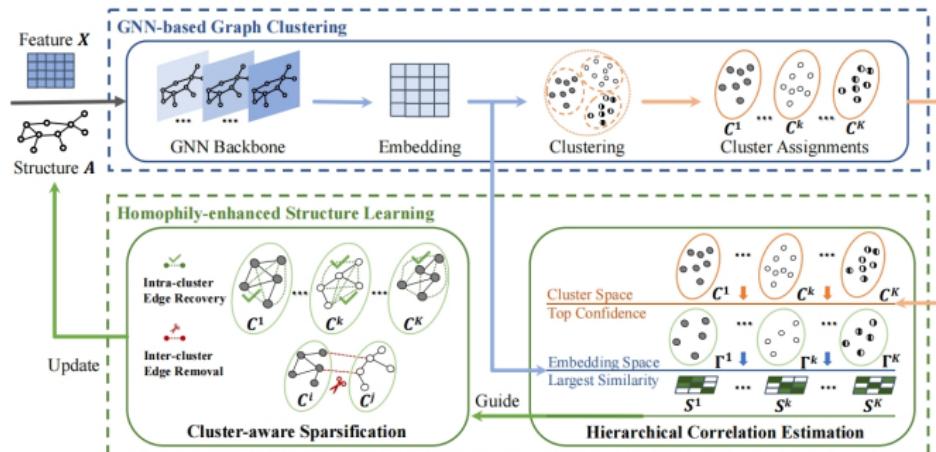
- ① 现有 GSL 方法大多是有监督的，而无监督方法未考虑图聚类问题的特性。
- ② 预实验发现提高图结构的 homophily 可以有效提高聚类效果。



¹¹[Gu et al., 2023]

HoLe: 模型架构

HoLE 采用类似 IDGL 的迭代更新机制，聚类和结构学习交替进行。



HoLe 方法示意图

每一个迭代中，HoLe 根据聚类结果，以同配性为目标更新结构。

HoLe: Homophily-enhanced Structure Learning

- ① 筛选每个类中的高置信度节点：

$$\Gamma^k = \left\{ v_i \mid c_i = k, \text{Rank}_{\{v_i \mid c_i = k\}}(q_{ik}) \leq \gamma * |\{v_i \mid c_i = k\}| \right\}$$

- ② 计算隐空间内的相似度： $S_{ij}^k = Z_i^k Z_j^{k^T} \in \mathbb{R}^{|\Gamma^k| \times |\Gamma^k|}$

- ③ 筛选相近同配边：

$$\mathcal{E}_{rc} = \bigcup_k \left\{ (v_i, v_j) \mid \text{Rank}(S_{ij}^k) \leq \xi * |\mathcal{E}| * \frac{N_k}{N}, \text{ and } c_i = c_j = k \right\}$$

- ④ 筛选不相似异质边：

$$\mathcal{E}_{rm} = \left\{ (v_i, v_j) \mid \text{Rank}(S_{ij}) \geq (1 - \eta) * |\mathcal{E}|, (v_i, v_j) \in \mathcal{E}, c_i \neq c_j \right\}$$

- ⑤ $A^* = A - A_{\mathcal{E}_{rm}} + A_{\mathcal{E}_{rc}}$

DSLR: Diversity Enhancement and Structure Learning for Rehearsal-based Graph Continual Learning¹²

使用 GSL 提升图连续学习场景下节点邻居信息的丰富程度

连续学习

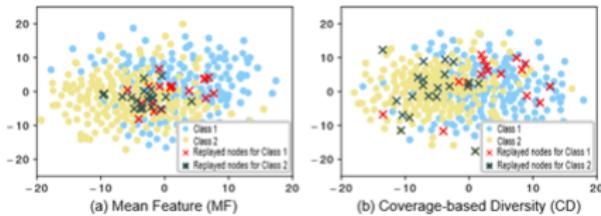
用于处理连续流式数据，旨在保留过去任务的知识以避免灾难性遗忘

该论文基于图连续学习中的 Rehearsal-based 方法：从先前数据中保留一部分节点作为 replay buffer，用于训练后续任务。主要目标为选择最佳的 replay buffer。

¹²[Choi et al., 2024]

DSLR: 动机

1. 现有方法采用不同类节点的 mean feature 来选择 replay buffer, 没有考虑类内 diversity



2. replay nodes 周围的邻居结构很重要, 使用 GSL 提高邻居质量

Table 1: Homophily ratio of the replayed nodes using MF & CD in Citeseer dataset.

	MF	CD
Class 1	0.68 ± 0.43	0.57 ± 0.45
Class 2	0.91 ± 0.24	0.92 ± 0.22
Class 3	0.82 ± 0.28	0.76 ± 0.40
Class 4	0.88 ± 0.26	0.82 ± 0.36

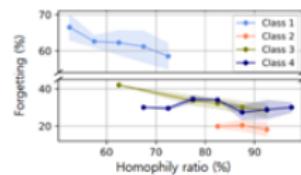
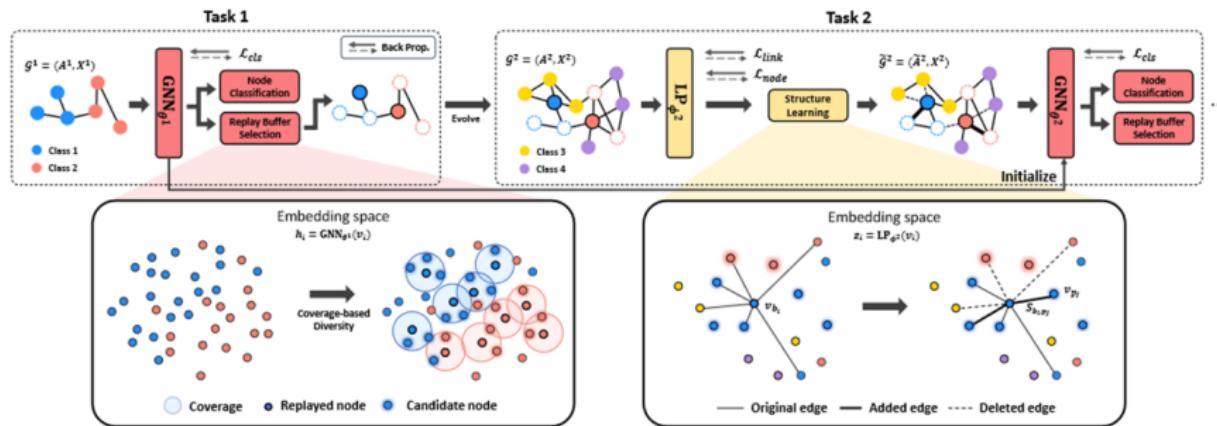


Figure 2: Forgetting over various homophily ratios of the replayed nodes in Citeseer dataset.

DSLR: 模型

针对 replay buffer 的 diversity 和邻居结构对应提出两个模块：

- ① 最大化表征空间中 replay nodes 周围一定距离内的节点数
- ② 从 GNN 模型得到的表征通过相似度计算边权，依赖 Link Predication Loss 与节点分类损失共同训练



DSLR 模型示意图

GSL 在其他领域的应用：小结

启发：作为一个有效的提高图结构质量的工具，GSL 可以用在更多其他图学习领域，从[数据中心](#)的角度（*a data-centric perspective*）解决其他领域的特有问题。

GSL Benchmark

对于这些 GSL 方法，我们思考以下几个问题：

- ① 这些 GSL 方法有效吗？
- ② 如果有效，这些 GSL 方法是什么起作用的？

近期出现了一些实验性工作尝试回答这些问题：

(NIPS 23, **Ours**) OpenGSL: A Comprehensive Benchmark for Graph Structure Learning

(NIPS 23) GSLB: The Graph Structure Learning Benchmark

(24) UGSL: A Unified Framework for Benchmarking Graph Structure Learning

OpenGSL: A Comprehensive Benchmark for Graph Structure Learning¹³

GSL 领域存在的问题

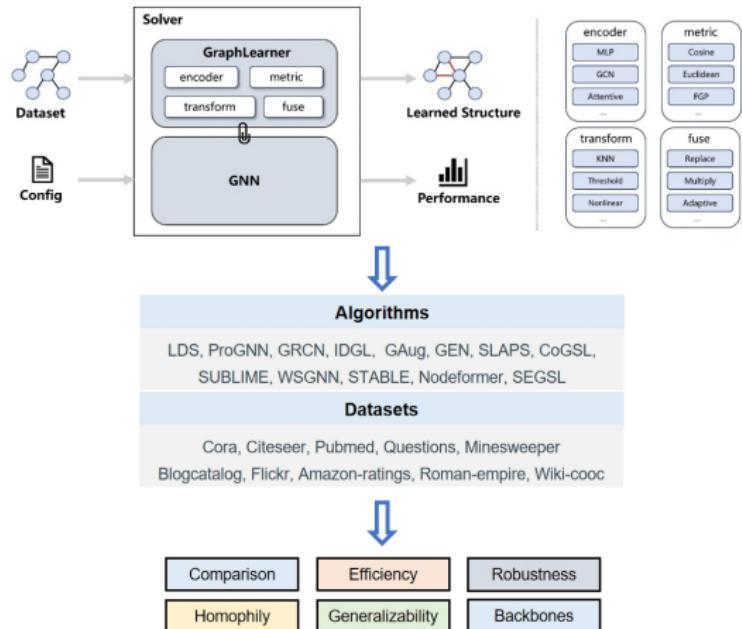
- ① 不同文章采用的实验设置（数据划分、处理等）不同
- ② 对学习的图结构缺少观察
- ③ 效率问题被忽视

OpenGSL 旨在建立一个详细的 benchmark 来解决上述问题。

¹³[Zhou et al., 2023]

OpenGSL: 介绍

- ① OpenGSL 是针对图结构学习的简单易用的开源包。
- ② 囊括了前沿的 GSL 方法，在统一设置下实现公平对比。
- ③ 对各 GSL 方法进行了多维度的分析。



Model	Cora	Citeseer	Pubmed	Questions	Minesweeper
GCN	81.95 ± 0.62	71.34 ± 0.48	78.98 ± 0.35	75.80 ± 0.51	78.28 ± 0.44
LDS	84.13 ± 0.52	75.16 ± 0.43	—	—	—
ProGNN	80.27 ± 0.48	71.35 ± 0.42	79.39 ± 0.29	—	51.43 ± 2.22
IDGL	84.19 ± 0.61	73.26 ± 0.53	82.78 ± 0.44	50.00 ± 0.00	50.00 ± 0.00
GRCN	84.61 ± 0.34	72.34 ± 0.73	79.30 ± 0.34	74.50 ± 0.84	72.57 ± 0.49
GAug	83.43 ± 0.53	72.79 ± 0.86	78.73 ± 0.77	—	77.93 ± 0.64
SLAPS	72.29 ± 1.01	70.00 ± 1.29	70.96 ± 0.99	—	50.89 ± 1.72
WSGNN	83.66 ± 0.30	71.15 ± 1.01	79.78 ± 0.35	—	67.91 ± 3.11
Nodeformer	78.81 ± 1.21	70.39 ± 2.04	78.38 ± 1.94	72.61 ± 2.29	77.29 ± 1.71
GEN	81.66 ± 0.91	73.21 ± 0.62	78.49 ± 3.98	—	79.56 ± 1.09
CoGSL	81.46 ± 0.88	72.94 ± 0.71	78.38 ± 0.41	—	—
SEGSL	81.04 ± 1.07	71.57 ± 0.40	79.26 ± 0.67	—	—
SUBLIME	83.33 ± 0.73	72.44 ± 0.89	80.56 ± 1.32	67.21 ± 0.99	49.93 ± 1.36
STABLE	83.25 ± 0.86	70.99 ± 1.19	81.46 ± 0.78	—	70.78 ± 0.27

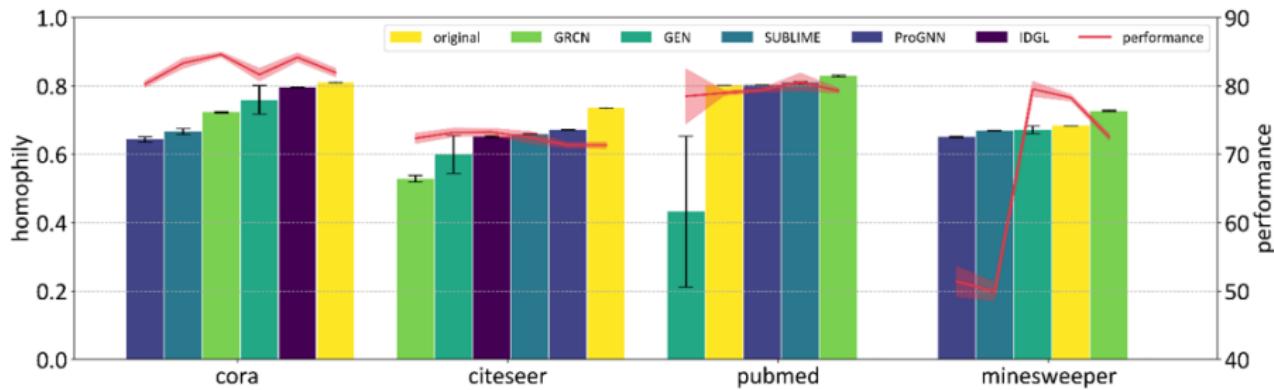
同配图上的分类表现

对于同配图，大部分 GSL 方法有提升作用；但是在类别不平衡的数据集上失效。

Model	BlogCatalog	Flickr	Amazon-ratings	Roman-empire	Wiki-cooc
GCN	76.12 ± 0.42	61.60 ± 0.49	45.24 ± 0.29	70.41 ± 0.47	92.03 ± 0.19
LDS	77.10 ± 0.27	—	—	—	—
ProGNN	73.38 ± 0.30	52.88 ± 0.76	—	56.21 ± 0.58	89.07 ± 5.59
IDGL	89.68 ± 0.24	86.03 ± 0.25	45.87 ± 0.58	47.10 ± 0.65	90.18 ± 0.27
GRCN	76.08 ± 0.27	59.31 ± 0.46	50.06 ± 0.38	44.41 ± 0.41	90.59 ± 0.37
GAug	76.92 ± 0.34	61.98 ± 0.67	48.42 ± 0.39	52.74 ± 0.48	91.30 ± 0.23
SLAPS	91.73 ± 0.40	83.92 ± 0.63	40.97 ± 0.45	65.35 ± 0.45	89.09 ± 0.54
WSGNN	92.30 ± 0.32	89.90 ± 0.19	42.36 ± 1.03	57.33 ± 0.69	90.10 ± 0.28
Nodeformer	44.53 ± 22.62	67.14 ± 6.77	41.33 ± 1.25	56.54 ± 3.73	54.83 ± 4.43
GEN	90.48 ± 0.99	84.84 ± 0.81	49.17 ± 0.68	—	91.15 ± 0.49
CoGSL	83.96 ± 0.54	75.10 ± 0.47	40.82 ± 0.13	46.52 ± 0.48	—
SeGSL	75.03 ± 0.28	60.59 ± 0.54	—	—	—
SUBLIME	95.29 ± 0.26	88.74 ± 0.29	44.49 ± 0.30	63.93 ± 0.27	76.10 ± 1.12
STABLE	71.84 ± 0.56	51.36 ± 1.24	48.36 ± 0.21	41.00 ± 1.18	80.46 ± 2.44

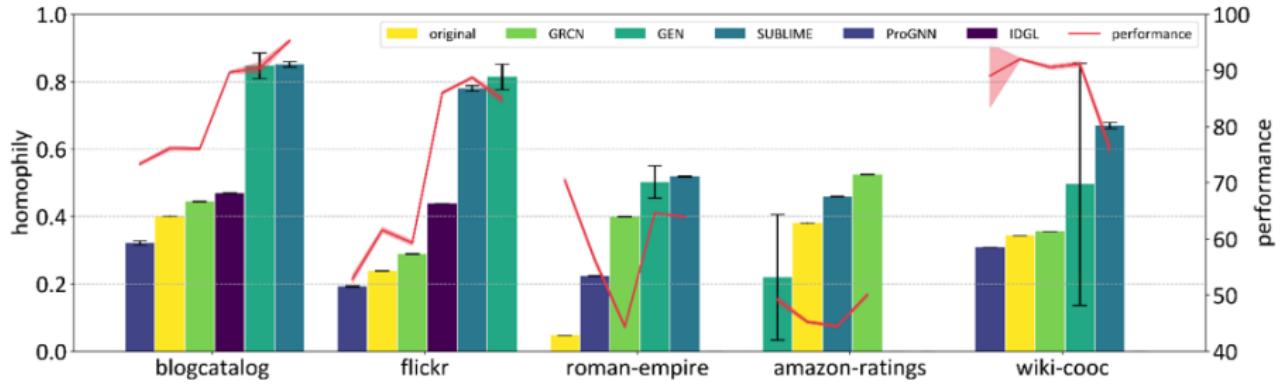
异配图上的分类表现

对于异配图，只有少数 GSL 方法在特定数据集上起作用。



同配图分类表现和学到结构的关系

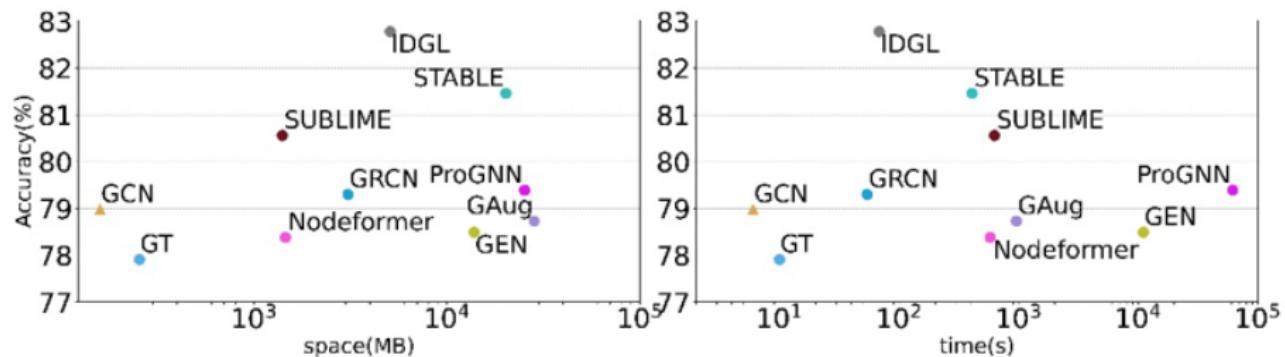
在异配图上，GSL 很容易提升结构的同配程度；在同配图上，学到结构的同配性很难进一步提升。



异配图分类表现和学到结构的关系

在大部分数据集上，分类表现和学到结构的同配性**无关**。

OpenGSL：效率分析



PubMed 上的各 GSL 方法效率

大部分 GSL 方法的时间和空间消耗大。

OpenGSL：未来研究

基于以上观察，有以下几个潜在的研究方向：

- ① 设计可以适应不同数据集（类别不平衡图、异配图）的 GSL 方法。
- ② 重新思考同配性的必要性，深入探索 GSL 的作用机制。
- ③ 发展可拓展的 GSL 方法以用于大规模数据集。



Paper :



Library :



课程总结

- ① 研究背景
- ② 基于优化的图结构学习
- ③ 基于度量学习的图结构学习
- ④ 基于生成模型的图结构学习
- ⑤ GSL 的最新研究进展
 - 无监督 GSL
 - 层次化 GSL
 - GSL 在其他领域的应用
 - GSL Benchmark

References I

-  Chen, Y., Wu, L., and Zaki, M. (2020). Iterative deep graph learning for graph neural networks: Better and robust node embeddings. *Advances in neural information processing systems*, 33:19314–19326.
-  Choi, S., Kim, W., Kim, S., In, Y., Kim, S., and Park, C. (2024). Dslr: Diversity enhancement and structure learning for rehearsal-based graph continual learning. In *Proceedings of the ACM on Web Conference 2024*, pages 733–744.

References II

-  Franceschi, L., Niepert, M., Pontil, M., and He, X. (2019). Learning discrete structures for graph neural networks. In *International conference on machine learning*, pages 1972–1982. PMLR.
-  Gu, M., Yang, G., Zhou, S., Ma, N., Chen, J., Tan, Q., Liu, M., and Bu, J. (2023). Homophily-enhanced structure learning for graph clustering. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 577–586.

References III

-  Jin, W., Ma, Y., Liu, X., Tang, X., Wang, S., and Tang, J. (2020).
Graph structure learning for robust graph neural networks.
In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 66–74.
-  Li, K., Liu, Y., Ao, X., Chi, J., Feng, J., Yang, H., and He, Q. (2022).
Reliable representations make a stronger defender: Unsupervised structure refinement for robust gnn.
In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 925–935.

References IV

-  Li, R., Wang, S., Zhu, F., and Huang, J. (2018). Adaptive graph convolutional neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.
-  Liu, Y., Zheng, Y., Zhang, D., Chen, H., Peng, H., and Pan, S. (2022). Towards unsupervised deep graph structure learning. In *Proceedings of the ACM Web Conference 2022*, pages 1392–1403.

References V

-  Luo, D., Cheng, W., Yu, W., Zong, B., Ni, J., Chen, H., and Zhang, X. (2021).
Learning to drop: Robust graph neural network via topological denoising.
In *Proceedings of the 14th ACM international conference on web search and data mining*, pages 779–787.
-  Wang, H., Fu, Y., Yu, T., Hu, L., Jiang, W., and Pu, S. (2023).
Prose: Graph structure learning via progressive strategy.
In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2337–2348.

References VI

-  Wang, R., Mou, S., Wang, X., Xiao, W., Ju, Q., Shi, C., and Xie, X. (2021).
Graph structure estimation neural networks.
In *Proceedings of the Web Conference 2021*, pages 342–353.
-  Yu, D., Zhang, R., Jiang, Z., Wu, Y., and Yang, Y. (2021).
Graph-revised convolutional network.
In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2020, Ghent, Belgium, September 14–18, 2020, Proceedings, Part III*, pages 378–393. Springer.

References VII

-  Zhou, Z., Zhou, S., Mao, B., Zhou, X., Chen, J., Tan, Q., Zha, D., Wang, C., Feng, Y., and Chen, C. (2023).
OpenGSL: A comprehensive benchmark for graph structure learning.
Advances in neural information processing systems.