

基于图互信息池化的分层图表示学习

吴辛尧^{1,2}, 徐计^{1*}

(1.省部共建公共大数据国家重点实验室(贵州大学), 贵州 贵阳 550025; 2.贵州大学计算机科学与技术学院, 贵州 贵阳 550025)

摘要:提出一种基于图互信息的池化算子——图互信息池化(graphical mutual information pooling, GMIPool)。GMIPool利用互信息神经估计度量节点及其对应的支撑图之间的图互信息(包括特征互信息和结构互信息),利用图互信息识别并保留图中的关键节点,构建更为紧凑的粗图。为确保原图和粗图在结构上的一致性,该方法利用节点之间的邻域关联性对粗图的结构进行修正。该方法在多个节点分类任务数据集上进行实验,验证了图互信息池化的有效性。

关键词:图神经网络;图池化;多粒度;图互信息;互信息神经估计

中图分类号:TP391 **文献标志码:**A

引用格式:吴辛尧,徐计.基于图互信息池化的分层图表示学习[J].山东大学学报(理学版),2025,60(7):84-93.

Hierarchical graph representation learning based on graphical mutual information pooling

WU Xinyao^{1,2}, XU Ji^{1*}

(1. State Key Laboratory of Public Big Data, Guizhou University, Guiyang 550025, Guizhou, China; 2. School of Computer Science and Technology, Guizhou University, Guiyang 550025, Guizhou, China)

Abstract: A graph pooling operator called graphical mutual information pooling (GMIPool). GMIPool is proposed utilizes mutual information neural estimation to measure the mutual information between nodes and their corresponding subgraphs, including both feature mutual information and structural mutual information. It leverages this information to identify and retain key nodes in the graph, constructing a more compact coarsened graph. To ensure structural consistency between the original and coarsened graphs, the method adjusts the coarsened graph's structure using the neighborhood correlations between nodes. Experiments on several node classification task datasets validate the effectiveness of GMIPool.

Key words: graph neural networks; graph pooling; multi-granularity; graphical mutual information; mutual information neural estimation

0 引言

卷积神经网络(convolutional neural networks, CNNs)^[1]在欧几里德数据空间上具有强大的多尺度局部空间特征提取能力,被广泛应用于图像分类、目标检测和语义分割等任务中。然而,现实世界中存在大量非欧几里德数据,如社交网络、化学分子和生物信息网络等,为处理这些不规则的数据,一些研究将卷积的思想推广到图结构数据。其中,图神经网络(graph neural networks, GNNs)^[2-3]基于消息传播和邻域特征聚合机制,能够有效地捕获图数据局部信息,实现高效的图分析和处理。目前,GNNs已在节点分类、链接预测、图结构学习、社区发现、图生成和图表示学习等各类图相关任务中得到广泛应用。

收稿日期:2024-05-16; 网络出版时间:2024-12-05 13:59:10

基金项目:国家自然科学基金资助项目(62366008, 61966005)

第一作者:吴辛尧(2001—),男,硕士研究生,研究方向为图神经网络。E-mail:gs.xinyaowu22@gzu.edu.cn

*通信作者:徐计(1979—),男,教授,硕士生导师,博士,研究方向为数据挖掘、粒计算和机器学习。E-mail:jixu@gzu.edu.cn

图数据上多粒度表示学习是对于图像多尺度特征学习的自然推广,可以有效提取图的分层表示。以图分类任务为例,该任务旨在利用节点特征和图拓扑信息预测整个图的标签,因此通常需要一个图级别的表示。然而,最初设计 GNNs 的目的是学习丰富的节点表示,因此常见的图表示通常由一个具有排列不变性质的函数聚合节点得到,例如求和或求平均。但是,上述方法可能无法区分图中不同子结构的内在属性以及节点在不同子结构中扮演的角色,导致学习到的图表示缺乏层次化特性,为获得图的层次化信息,须要实施逐层的池化过程。

最近,研究者们提出一些基于数据驱动的图池化方法^[4-5],主要分为节点聚类 and 节点丢弃。节点聚类将一组细粒度节点聚合成一个较粗粒度的节点,而节点丢弃则根据节点的重要性删除不必要的节点和相应的边,得到原图的诱导子图。然而,节点聚类方法需要额外的聚类模块,时间复杂度较高,且粗化后的节点与原始节点可能缺乏直接关联,不利于不同粒层的信息交互。节点丢弃方法生成的子图可能无法保留底层的子结构,失去图拓扑信息的完整性,并且通常无法充分考虑节点的局部信息。

针对上述挑战,本文提出图互信息池化(graphical mutual information pooling, GMIPool)。GMIPool 利用节点邻域结构为其构建支撑图并遵循 infomax^[6]准则从特征和结构两方面度量节点潜在表示和支撑图之间的图互信息(graphical mutual information, GMI),精准地识别并保留那些能最大程度表征其支撑图(即原图中的局部子结构)的关键节点,从而构建信息丰富的粗图。GMIPool 不仅直接建立不同粒层上节点之间的对应关系,有利于实现跨粒层信息融合,而且基于支撑图,GMIPool 充分考虑了图的局部信息,有效避免因忽视局部结构而导致的信息丢失。此外,GMIPool 利用节点邻域关联性对粗图结构进行修正,保持了不同粒度层间图的拓扑结构一致性。在多个公开节点分类任务数据集上的实验结果表明,GMIPool 显著提升了分类性能,验证了其有效减少信息丢失、保持结构一致性的能力。

1 相关工作

1.1 互信息与互信息神经估计

互信息(mutual information, MI)衡量两个随机变量共享的信息量,不同于普通的相似性度量方法,它能捕捉到变量间的非线性统计相关性,因此是一种更为全面的度量两个随机变量依赖性和相关性的方法。尽管互信息具有许多良好的特性,但在实际应用中,特别是对于高维连续变量,互信息的估计面临着巨大的困难。这些困难包括数据稀疏性导致的估计不准确,以及复杂的计算需求和概率分布的精确性的高要求。为解决这些问题,研究者引入了互信息神经估计(mutual information neural estimation, MINE),该方法通过训练一个统计网络作为鉴别器,计算两个随机变量互信息的紧致下界估计互信息,该下界通过区分从数据中采样的正样本(来自两个随机变量的联合概率分布)和负样本(来自两个随机变量的边缘分布乘积)获得。不断提高该下界的值,不仅能间接地最大化两个随机变量之间的互信息,还能得到其近似值。在图表示学习领域,基于 infomax 准则,互信息被广泛应用于图数据的表征学习中。Hjelm 等^[7]通过最大化深度神经网络编码器输入和输出层之间的互信息首次提出深度信息最大化(deep infomax, DIM)。Fedus 等^[8]最大化图中每个节点表示和图表示之间的互信息,首次将 DIM 的概念引入到图领域,提出深度图信息最大化(deep graph infomax, DGI)。Peng 等^[9]从节点特征和拓扑结构两个方面衡量互信息,提出图互信息(graphical mutual information, GMI)衡量输入图和节点嵌入之间的关联性。Pang 等^[10]提出粗图信息最大化池化(coarsened graph infomax pooling, CGIPool)最大化池化前后图表示之间的互信息进行图分类。

1.2 图池化

图池化是学习多粒度图表示的一种有效方法,受到广泛的关注。为了生成任意图的嵌入表示,有学者提出了全局池化方法。例如,Vinyals 等^[11]通过长短期记忆网络(long short-term memory, LSTMs)聚合节点信息以获取图的全局表示。但这类方法不能获取图中的层次信息,为解决这一问题,Zhang 等^[12]提出一种可微排序的池化方法,该方法对节点的特征映射进行降序排序,并利用前 k 个节点来表示图。受聚类方法的启发,Ying 等^[13]提出名为可微分池化(differentiable pooling, DiffPool)的池化算子,该方法使用神经网络来学习一个软分配矩阵,将节点映射到类簇集合,从而产生粗粒度的图。Yuan 等^[14]通过显式地捕捉高阶结构关系对 DiffPool 进行拓展。Ma 等^[15]基于谱聚类将图分割成子图进行池化。然而,由于其中涉及聚类过程,这些方法往往具有较高的时间复杂度。相比之下,另一类采用丢弃图中不重要节点的策略进行池化的方法效率更高。Gao 等^[16]

提出名为图池化(graph pooling, gPool)的池化算子, gPool 利用全局可训练投影向量将节点特征映射为标量, 实现全局节点选择。Lee 等^[17]基于节点的自注意力机制, 提出自注意力图池化(self-attention graph pooling, SAGPool)。Gao 等^[18]基于信息增益准则提出一种非参数化的节点选择方案。Zhang 等^[19]基于注意力融合机制提出多视图的图池化(multi-view graph pooling, MVPool)相较于仅从单一视图选择节点的方法(如 SAGPool 和 gPool), 其池化过程更具鲁棒性。Chen 等^[20]提出节点信息最大化池化(vertex infomax pooling, VIPool), 该方法最大化节点及其邻域的互信息选择最能代表邻域的节点进行池化。

上述方法中, gPool 倾向于选择具有某种特定模式的节点, 并且使用全局投影向量可能导致局部信息的丢失。由于相同标签的邻域节点注意力分数较大, SAGPool 选择的节点往往集中在某些特定邻域, 导致模型忽略图中的一些重要局部信息。VIPool 最大化节点和其邻域的特征层面的互信息, 忽略了图拓扑结构的互信息。GMIPool 与上述方法同属于节点丢弃类的池化方法, 但是从信息论的角度更具可解释性。GMIPool 以某一节点作为中心为其构建支撑图并估计节点和支撑图之间的 GMI, 更好地理解图结构中的局部和全局信息。相比上述方法, GMIPool 衡量的互信息更加全面, 可以更好地捕获图中的重要信息, 为后续任务提供更有价值的输入。

2 基于图互信息的池化

2.1 符号定义

给定一个图 $G = \{V, E\}$, 其中 V 表示节点集合, E 是节点之间边的集合。令 $A \in \mathbf{R}^{n \times n}$ 为图的邻接矩阵, $n = |V|$ 为节点个数, $X \in \mathbf{R}^{n \times d_0}$ 为节点的特征, d_l 为第 l 层的节点特征维度。由于池化操作会改变图结构和节点个数, G_l 表示第 l 层的输入图, $A_l \in \mathbf{R}^{n_l \times n_l}$ 和 $X_l \in \mathbf{R}^{n_l \times d_l}$ 分别表示输入图 G_l 的邻接矩阵和节点特征, $n_l = |V_l|$ 为输入图 G_l 中节点个数, $H_l \in \mathbf{R}^{n_l \times d}$ 表示输入图经 GNNs 编码后的潜在表示。此外, 令 G_l 中节点 v_i^l 的支撑图为 G_l^i , 对应的邻接矩阵和特征向量为 A_l^i 和 X_l^i , n_l^i 为支撑图 G_l^i 中节点个数。

2.2 图互信息池化

利用 GMIPool 为当前层输入图 G_l 中每个节点 v_i^l 构建一个对应的支撑图 $G_l^i = \{A_l^i, X_l^i\}$, 然后, 用 MINE 最大化节点的潜在表示 h_i^l 和支撑图 $G_l^i = \{A_l^i, X_l^i\}$ 之间的互信息 $I(h_i^l, G_l^i)$, 在 G_l 中选择互信息较大的节点子集形成粗粒度图。 $I(h_i^l, G_l^i)$ 包含特征互信息和结构互信息。

对于特征互信息 $I(h_i^l, X_l^i)$, 令 $p(X_l^i)$ 、 $p(h_i^l)$ 分别为 X_l^i 、 h_i^l 的概率分布, $p(h_i^l, X_l^i)$ 联合概率分布, 基于信息论, $I(h_i^l, X_l^i)$ 的定义为

$$I(h_i^l, X_l^i) = \int_{\mathcal{H}} \int_{\mathcal{X}} p(h_i^l, X_l^i) \log \frac{p(h_i^l, X_l^i)}{p(h_i^l)p(X_l^i)} dh_i^l dX_l^i, \quad (1)$$

式中 \mathcal{H} 、 \mathcal{X} 分别为 h_i^l 、 X_l^i 的向量空间。

然而直接使用互信息神经估计计算 $I(h_i^l, X_l^i)$ 是不切实际的, 因为互信息神经估计中的鉴别器仅接受固定尺寸的输入, 而不同支撑图 G_l^i 的节点个数不同, 导致鉴别器接受的输入尺寸是变化的, 但 $I(h_i^l, X_l^i)$ 可以分解为节点 v_i^l 的 h_i^l 、 G_l^i 中节点特征之间局部互信息的加权和, 即

$$I(h_i^l, X_l^i) = \sum_j^{n_l^i} w_{i,j} I(h_i^l, x_{l,i}^j), \quad \text{s.t.} \quad \frac{1}{n_l^i} < w_{i,j} \leq 1, \quad (2)$$

式中: $x_{l,i}^j$ 是 G_l^i 中第 j 个节点特征向量, n_l^i 表示 G_l^i 节点个数, $w_{i,j}$ 为权重衡量局部互信息对于全局互信息的贡献。证明参阅文献[9]。通过上述分解, 利用互信息神经估计中的鉴别器估计局部互信息 $I(h_i^l; x_{l,i}^j)$, 然后进行加权求和即可得到全局特征互信息 $I(h_i^l, X_l^i)$ 。 $w_{i,j}$ 的精确值难以获取, GMIPool 中将 $w_{i,j}$ 设置为 $\frac{1}{n_l^i}$ 。这样最大化 $\sum \frac{1}{n_l^i} I(h_i^l; x_{l,i}^j)$ 等价于最大化 $I(h_i^l, X_l^i)$ 的下界, 使得 $I(h_i^l, X_l^i)$ 在一定程度上也能最大化。 GMIPool 更强调最大化节点潜在嵌入 h_i^l 和支撑图节点的输入特征 X_l^i 之间的互信息, 当 GMIPool 中鉴别器无法准确计算互信息

时,节点潜在表示 \mathbf{h}_i^j 仍然可以包含大量原始图的信息,进一步池化后得到的粗图也能尽可能地保留原图的信息。为最大化 $I(\mathbf{h}_i^j, \mathbf{X}_i^j)$, GMIPool 使用 Jensen-Shannon 的互信息估计器,即

$$I(\mathbf{h}_i^j; \mathbf{x}_{l,i}^j) = -\log(1 + \exp(-D_\theta(\mathbf{h}_i^j, \mathbf{x}_{l,i}^j))) - \mathbf{E}_{\tilde{\mathbf{P}}}[\log(1 + \exp(D_\theta(\mathbf{h}_i^j, \mathbf{x}_{l,i}^j)))] \quad (3)$$

式中: D_θ 是参数为 θ 的统计网络,用于衡量 \mathbf{h}_i^j 和 $\mathbf{x}_{l,i}^j$ 之间的互信息, $\tilde{\mathbf{P}}$ 是负样本空间, $\mathbf{x}_{l,i}^j$ 是从 $\tilde{\mathbf{P}}$ 中采样的负样本。在 GMIPool 中可以使用任意输出为标量的统计网络估计 $I(\mathbf{h}_i^j; \mathbf{x}_{l,i}^j)$, 为保持模型的简洁性, GMIPool 使用一个简单的双线性层作为统计网络,即

$$D_\theta(\mathbf{h}_i^j; \mathbf{x}_{l,i}^j) = \sigma(\mathbf{h}_i^{jT} \Theta \mathbf{x}_{l,i}^j), \quad (4)$$

式中: σ 是激活函数, Θ 是 D_θ 的参数。

对于结构互信息, GMIPool 利用节点潜在表示的相似性重构图结构,使用交叉熵估计重构图 \mathbf{S}_l 和原图 \mathbf{A}_l 之间的互信息,结构互信息为

$$\mathbf{S}_l = \sigma(\mathbf{H}_l \mathbf{H}_l^T) \quad (5)$$

$$I(s_l^{i,j}, a_l^{i,j}) = a_l^{i,j} \log s_l^{i,j} + (1 - a_l^{i,j}) \log(1 - s_l^{i,j}),$$

式中: \mathbf{S}_l 是重构后图结构, σ 是激活函数将节点之间的相似性投影到 $[0, 1]$ 之间。 $s_l^{i,j}$ 和 $a_l^{i,j}$ 分别为重构图 \mathbf{S}_l 与原图 \mathbf{A}_l 中边的权重。通过最大化 $I(s_l^{i,j}, a_l^{i,j})$, GMIPool 使得到的节点嵌入能够符合原始图的拓扑关系中,节点潜在表示包含更多的结构信息。

由上述特征互信息和结构互信息,得到节点嵌入 \mathbf{h}_i^j 和支撑图 G_i^j 的图互信息

$$I(\mathbf{h}_i^j, G_i^j) = \sum \frac{1}{n_i} I(\mathbf{h}_i^j, \mathbf{x}_{l,i}^j) + I(s_l^{i,j}, a_l^{i,j}). \quad (6)$$

图1为 GMIPool 如何利用图互信息进行池化的具体过程。以每个节点的一阶邻域作为支撑图为例,在池化前, GMIPool 通过保持图结构随机打乱节点的方式生成图的负样本,根据邻域关系为正负样本中的节点生成对应的支撑图,利用权重共享的双线性层作为 MINE 的鉴别器,估计节点和支撑图之间的 GMI。 GMIPool 能有效地量化节点与其周围环境之间的关联程度,从而更全面地理解图结构中的局部和全局信息。池化时, GMIPool 采用贪心策略直接将互信息视为节点的重要性评分,保留互信息较大的前 $n_{l+1} = m_l$ 个节点 (r 为池化率) 到粗图中,以确保关键节点的信息被充分保留,即

$$p = D_\theta(\mathbf{h}_i^j; \mathbf{X}_i^j), \quad (7)$$

$$\Omega = T_{\text{topk}}(p, n_{l+1}), \quad (8)$$

式(7)中, p 是节点关于支撑图的互信息,式(8)中 T_{topk} 是全局排序函数, Ω 为基于互信息所选的关键节点索引的集合。值得注意的是, GMIPool 仅利用特征互信息进行节点选择,这是因为 $I(s_l^{i,j}, a_l^{i,j})$ 确保 GNNs 输出的节点嵌入符合原图的拓扑结构中,原图的拓扑信息已经被纳入节点嵌入中。

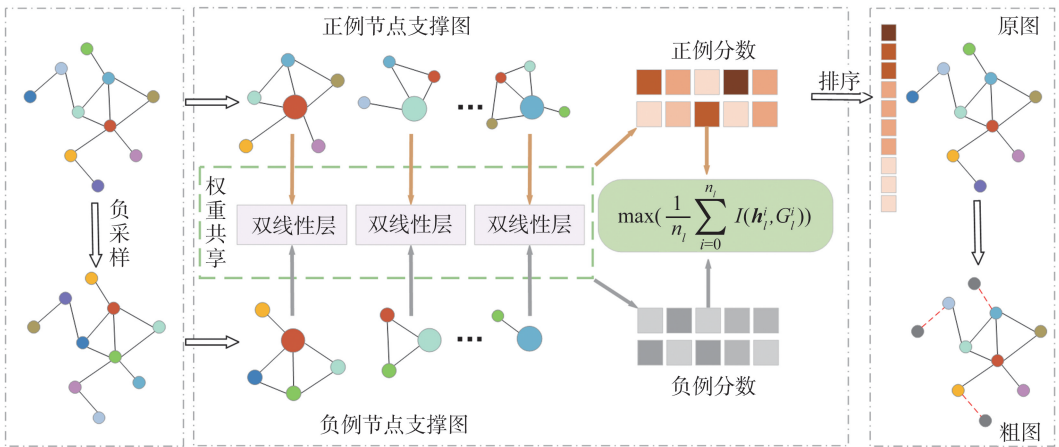


图1 GMIPool 的示意图(粗图中灰色节点为丢弃的节点,虚线表示丢弃的边)

Fig.1 Illustration of GMIPool (In the coarsened graph, gray nodes represent the discarded nodes, and dashed lines indicate the discarded edges.)

通过上述池化操作,可以提取原图中的关键节点。接下来,利用这些节点以及原图的邻接矩阵构建粗图,粗图的构建涉及节点特征和邻接矩阵。

使用节点的重要性评分和节点潜在表示构建方法为

$$\mathbf{X}_{l+1} = \mathbf{H}_l[\Omega, :] \odot p(\Omega)^T \mathbf{I}, \quad (9)$$

式中: \odot 为哈达玛积, $\mathbf{I} \in \mathbf{R}^{n_i \times n_i}$ 是全 1 的列向量, $\mathbf{H}_l[\Omega, :]$ 为从 Ω 中得到关键节点特征。粗图不仅保留原图中节点的重要性关系, 还能使梯度在网络的各层中传播, 从而实现端到端的模型训练。

直接利用节点及其对应的边构建粗图的不仅导致大量拓扑信息丢失, 而且使粗图与原图之间缺乏结构一致性, 这不利于下一次池化时结构互信息估计。为确保池化前后的结构一致性, 利用选定节点之间的邻域关系修正粗图结构。具体的构建方法为

$$(\mathbf{A}_{l+1})_{i,j} = (\mathbf{A}_l[\Omega, \Omega])_{i,j} + \lambda \frac{2|\mathcal{N}_i^i \cap \mathcal{N}_j^j|}{|\mathcal{N}_i^i \cup \mathcal{N}_j^j|}, \quad (10)$$

式中: \mathbf{A}_l 和 \mathbf{A}_{l+1} 分别是原图和粗图的邻接矩阵, \mathcal{N}_i^i 是 \mathbf{A}_l 中包含节点 v_i 的邻域节点集合, 为调节重构过程对粗图的影响, 引入超参数 λ , $\mathbf{A}_l[\Omega, \Omega]$ 为利用 Ω 从原图中抽取的子图。若原图中的两个节点没有直接连接, 但节点之间存在许多共同的邻居节点, 那么在粗图中这 2 个节点应该更相似, 直接使用关键节点构建的粗图导致上述结构信息丢失。此外, 由于节点之间没有直接的边相连, 会进一步阻碍后续 GNNs 中的消息传播。通过对粗图的结构修正, 自适应地调整图结构以适应下游任务。

2.3 上采样层

池化提取不同粒层的图结构特征, 而上采样则是将不同粒层的图恢复到原始的图结构中, 促进不同粒层之间的特征交互, 使图中不同层级的信息能够被充分利用, 上采样是池化的逆操作。

为将粗图恢复到原图, 记录 GMIPool 池化时选择的节点位置 Ω 和潜在表示 \mathbf{H}_l , 并使用这些信息将对应的节点放回到原图中。令 \mathbf{A}_{l+1} 和 \mathbf{H}_{l+1} 为原图的邻接矩阵和节点特征, 上采样通过下述公式融合粗粒度图和原图信息:

$$\begin{aligned} \mathbf{F}_{l+1} &= \mathbf{O}, \quad \mathbf{F}_{l+1}[\Omega] = \mathbf{H}_l, \\ \mathbf{H}_{l+1} &= E_{\text{GNNs}}(\mathbf{F}_{l+1}, \mathbf{A}_{M-l+1}) + \mathbf{H}_{M-l+1}, \end{aligned} \quad (11)$$

式中: $\mathbf{O} \in \mathbf{R}^{n_i \times d}$ 是零矩阵, M 为模型最大层数, l 为模型当前层数, E_{GNNs} 是图卷积层构成的编码函数, 当上一层的节点特征映射回原位置后, 利用该函数融合不同粒层特征。

2.4 损失函数和训练策略

为有效训练模型, 从池化损失和分类损失 2 个方面考虑训练损失。池化损失是计算 GMI 并选择关键节点, 而分类损失则用于执行下游分类任务。

在训练时, 引入一个超参数 β 以更好的平衡学习过程中的特征互信息和结构互信息, 池化损失为

$$L_{\text{pool}} = -\frac{1}{n_i} \sum I(\mathbf{h}_i^i, G_i^i) = -\left(\frac{1}{n_i} \sum I(\mathbf{h}_i^i; \mathbf{X}_i^i) + \beta I(s_i^{i,j}, a_i^{i,j}) \right), \quad (12)$$

式中 n_i 为当前输入图 G_i 中节点数目。实验任务是节点分类, 因此选用交叉熵作为分类损失函数, 即

$$L_{\text{task}} = -\sum_i (y_i \log(\hat{y}_i) + (1-y_i) \log(1-\hat{y}_i)), \quad (13)$$

式中: y_i 是节点 i 所属的类别标签, \hat{y}_i 是模型的预测概率分布。总损失为

$$L = \alpha L_{\text{pool}} + L_{\text{task}}, \quad (14)$$

式中超参数 α 用于调节模型对池化损失和分类损失的重视程度, 进而影响模型在学习过程中对于图互信息和下游任务的优化方向。在模型训练的早期, 设置较大的 α , 随着训练次数的增加 α 降低, 使得 GMIPool 在训练初期会更加关注互信息评估, 加快鉴别器的收敛速度, 有助于在较早阶段获得更精确的互信息评估值, 进而准确地选择出关键节点。随着训练的迭代, 模型逐渐进入到下游任务, 有助于避免模型过早陷入局部最优解, 从而提高模型的稳健性和泛化能力。

2.5 总体框架

图 2 为带有 GMIPool 池化层的节点分类任务模型的整体架构。该模型采用编码器-解码器的设计, 编码器由多个卷积层和 GMIPool 池化层堆叠而成, 用于提取不同粒度层级的图信息。解码器由上采样层和卷积层组成, 用于恢复图结构并将不同粒度层级的图信息融合到原始图中, 从而生成信息丰富的节点嵌入。最终, 基于生成的节点嵌入进行节点分类任务。实验中, 本架构使用图卷积网络^[21] (graph convolutional networks, GCNs)

作为卷积层。

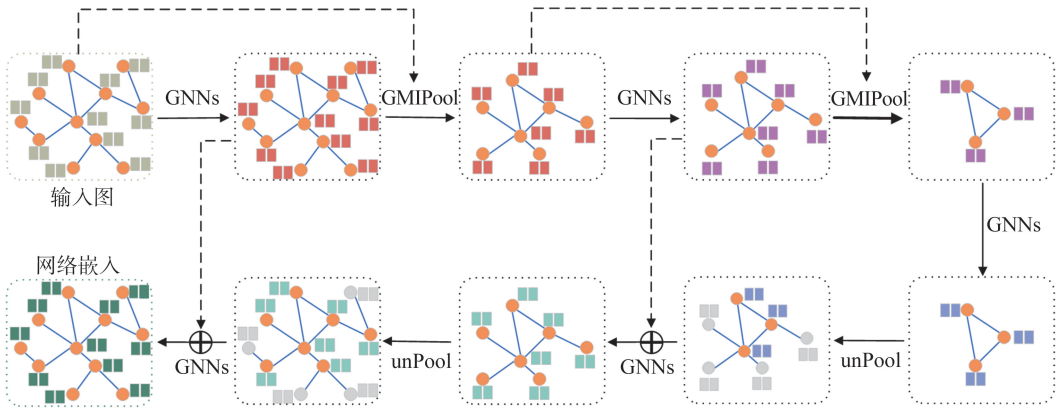


图 2 基于 GMIPool 的图表示学习框架示意图

Fig.2 Illustration of the Graph Representation Learning Framework based on GMIPool

3 实验

3.1 数据集

本文在多个数据集上进行直推式半监督节点分类任务。表 1 为实验中所用数据集的统计数据, 这些数据集包括 3 个常用的引文基准数据集: Cora、Citeseer、Pubmed, 2 个广泛使用的合著者数据集: Coauthor-CS 和 Coauthor-Physics。在引文数据集中, 节点表示文档, 边表示文档之间的引用关系; 在合著者数据集中, 节点表示作者, 边表示作者之间的合著关系。在直推式半监督节点分类任务中, 模型可以访问所有节点的属性信息和完整的图结构, 但是只有少部分节点标签可以使用。实验过程严格按照之前的研究中的实验设置: 对于引文数据集, 遵循文献[21]中标准划分数据集。对于合著数据集, 每个类随机取 20 个节点作为训练集, 30 个节点作为验证集, 其余节点为测试集。

表 1 节点分类数据集的统计数据
Table 1 Node classification dataset statistics

数据集	节点数	特征维度	类别	数据集节点划分(训练/验证/测试)
Cora	2 708	1 433	7	140/500/1 000
Citeseer	3 327	3 703	6	120/500/1 000
Pubmed	19 717	500	3	60/500/1 000
Coauthor-CS	18 333	6 805	15	300/450/17 583
Coauthor-Physics	34 493	8 415	5	100/150/34 243

3.2 对比基线和实验设置

在介绍实验设置前, 简要介绍实验中使用的节点分类基线模型。总体而言, 这些基线模型分为传统图嵌入模型、半监督图卷积模型以及分层表示模型 3 个类别。

传统图嵌入模型。实验选取 2 个具有代表性的模型作为传统图嵌入模型的基线模型, 分别为多层感知机 (multilayer perceptron, MLP) 和深度游走 (deep walk), 其中, MLP 将节点特征投影到潜在向量空间进行节点分类任务, 而 Deep Walk 则通过随机游走探索图结构, 学习节点嵌入。

半监督图卷积模型。这类模型通过在图上执行卷积操作来生成节点嵌入, 但由于缺乏池化操作, 忽略了图中的层级结构。选择 3 种经典的半监督图卷积模型作为基线模型, 分别是图卷积网络^[21] (graph convolutional networks, GCNs)、图注意力网络^[22] (graph attention networks, GAT) 和图采样与聚合网络^[23] (graph sample and aggregate, GraphSAGE)。

分层表示模型。分层表示模型将图池化操作嵌入到 GNNs 中, 使模型能有效学习到图的分层表示。实验中, 选择图 U 型网络^[16] (graph-U-Nets, g-U-Nets)、多视图池化^[19] (multi-view graph pooling, MVPool) 和自适应多粒度图神经网络^[24] (adaptive multi-grained graph neural networks, AdamGNNs) 作为主要的对比模型。

g-U-Nets和MVPool在节点分类模型上采用与GMIPool相同的架构,但它们的池化方法有所不同。g-U-Nets使用全局投影向量进行池化,MVPool利用注意力机制融合特征、结构等不同视角的重要性进行池化,AdamGNN利用自网络实现自适应池化。

实验设置:GMIPool基于Pytorch Geometric实现,代码可在<https://github.com/XinYaoWu111/GMIPool>中获取。对原论文中已经测试过的数据集,实验中直接引用其结果,原论文中没有测试过的部分数据集,本实验严格遵循原论文的实验方法进行对比实验。在GMIPool的具体实现中,使用Adam的优化器并将dropout机制用于节点特征矩阵,丢失率为0.76。实验中使用网格搜索寻找最优的参数组合,学习率搜索范围为[0.1,0.05,0.01,0.001],权重衰减搜索范围为[0.1,0.01,0.001]。对于所有的数据集,GMIPool中所有层的节点潜在表示维度均为32,池化层数为2~4层,每层的池化率为0.2~0.8。

3.3 实验结果

本节对比GMIPool和各基线模型在节点分类任务上的性能,实验中使用节点分类准确率作为度量模型性能的指标,节点分类准确率为

$$P = \frac{1}{N} \sum_{i=1}^N I(y_i = \hat{y}_i), \quad (15)$$

式中: N 为节点个数, I 为指示函数, y_i 是节点真实标签, \hat{y}_i 是模型预测标签。

为确保基线模型的可靠性和鲁棒性,实验中选取验证集上分类损失最低的模型进行评估,并对各基线模型进行10次重复评估,计算平均分类准确率及其标准差。表2概括了各模型在不同数据集上,节点分类任务的分类准确率以及标准差,其中加粗的数据表示该模型在对应的数据集上分类准确率最高,下划线表示次优,表2中最后一行是GMIPool相较于当前最先进模型在各数据集上分类准确率的提升百分比,以量化其性能优势。

表2 各模型在不同数据集上的准确率
Table 2 Accuracy (%) of different algorithms on various datasets

基线模型	Cora	Citeseer	Pubmed	Coauthor-CS	Coauthor-Physics
MLP	56.8±0.9	56.7±1	69.7±3.3	86.3±1.2	87.9±1.6
Deep Walk	67.2±0.8	41.2±2.0	63.7±1.9	78.2±1.5	76.6±1.6
GCN	81.5±0.6	70.0±0.7	79.0±0.6	88.0±1.1	92.1±1.2
GAT	83.1±0.7	72.5±0.7	79.0±0.3	90.2±0.6	90.7±1.7
GraphSage	79.4±1.8	66.8±3.5	75.8±1.0	90.6±0.5	90.9±1.7
AdamGNN	83.1±0.9	72.8±0.5	<u>80.6±0.6</u>	91.5±0.2	93.1±0.5
MVpool	<u>85.0±0.4</u>	<u>73.5±0.5</u>	80.5±0.4	92.0±0.5	<u>94.0±0.6</u>
g-U-Nets	84.4±0.6	73.2±0.5	79.6±0.2	90.9±0.5	93.1±0.8
GMIPool	85.4±0.2	74.1±0.3	81.1±0.4	<u>91.9±0.1</u>	94.5±0.1
相对提升	0.51	0.78	0.62	-0.11	0.53

从表2中可以观察到传统的图嵌入模型分类准确率相对较低,因为传统的图嵌入模型未能充分的对图数据进行编码。其中,MLP仅使用节点特征信息而忽略图中的丰富的结构信息,相对的,而DeepWalk仅依赖图的结构信息获取节点嵌入,忽略了节点特征的重要性。相较于MLP和DeepWalk等,半监督图卷积从节点及其邻域中迭代地聚合信息将节点特征和图结构编码到统一的表示空间中,通过这种灵活的融合方式,这类模型在各数据集上都取得较大的性能提升,但由于缺乏池化操作,无法提取图中的层次信息。为提取图数据中的层次结构,提出了MVPool和g-U-Nets等分层图表示模型,这类模型分类准确率在各数据集上都不同程度的超过半监督图卷积模型。这归功于池化操作提取出不同粒层的图结构并通过上采样操作进一步融合这些层次结构信息,得到更多语义信息节点表示。GMIPool属于分层表示模型的范畴,GMIPool的分类准确率相比于g-U-Nets和MVPool等模型更高。g-U-Nets使用一个全局投影向量对节点特征进行重要性评分,并依据这一评分进行池化,这会导致模型选择节点时倾向于选择具有某种特定模式的节点,且使用全局投影向量可能导致局部信息的丢失,而GMIPool估计节点和支撑图之间的图互信息,可以更好地理解图结构中的局部和全局信息。MVPool利用注意力机制融合节点的特征、结构等多个视角的重要性进行池化,但是仍然不能有效保留关键节点信息到粗粒图中,相对地GMIPool从信息论角度更具可解释性,可以有效降低池化过程中信息丢失。

整体而言,GMIPool在多个基准数据集上相较于当前最先进的方法均取得了显著提升,特别是在Cora、

Pubmed、Citeseer、Physics 等数据集上分类准确率分别相对提高了 0.51%、0.78%、0.62% 和 0.53%,并且在数据集 CS 上仅比最优方法低 0.1%。此外,相较于没有池化层的普通 GCN, GMIPool 的加入使得分类性能在所有数据集上均有显著提升,分类准确率平均提升 3.28%。这些结果表明,GMIPool 能够更有效地提取和利用节点的重要性信息,从而在节点分类任务中表现出更强的效果,验证了该方法的有效性。

3.4 可视化分析

可视化分析对于理解和评估模型的性能非常重要,因为它直观地展示了不同模型在区分节点类别上的能力。为进一步验证 GMIPool 在节点分类任务中的有效性和优越性,本实验在 Cora 上使用 t-SNE 算法对各模型生成的节点嵌入进行可视化分析。t-SNE 是一种常用的可视化方法,它通过保持高维空间中的局部结构,将高维数据映射到二维或三维空间中。

如图 3 所示,图中相同的颜色的点表示它们属于同一类别。图 3(a) 为原图中节点嵌入的 t-SNE 可视化结果,可以观察到原始节点嵌入的 t-SNE 图较为杂乱,这说明原始的节点嵌入在高维空间中不存在明显的类别关系。图 3(b)、(c)、(d) 分别为经 GCN、g-U-Nets 和 GMIPool 编码后节点嵌入的 t-SNE 可视化图。GCN 编码后不同类别的节点虽然形成了不同的类簇,但类簇间的边界并不清晰;g-U-Nets 编码后的节点嵌入有所改善,不同类别节点不仅形成了簇,且部分簇之间存在明显边界。图 3(d) 为 GMIPool 的结果,不同类别的节点不仅形成了明显的簇,而且类簇之间有清晰的边界,这表明 GMIPool 在嵌入空间中能更好地捕捉和放大不同类别节点之间的差异,同时压缩同类别节点之间的相似性,从而增强节点分类的效果,显示了 GMIPool 捕捉和放大类别差异方面的优势,使得分类任务更加准确和可靠。

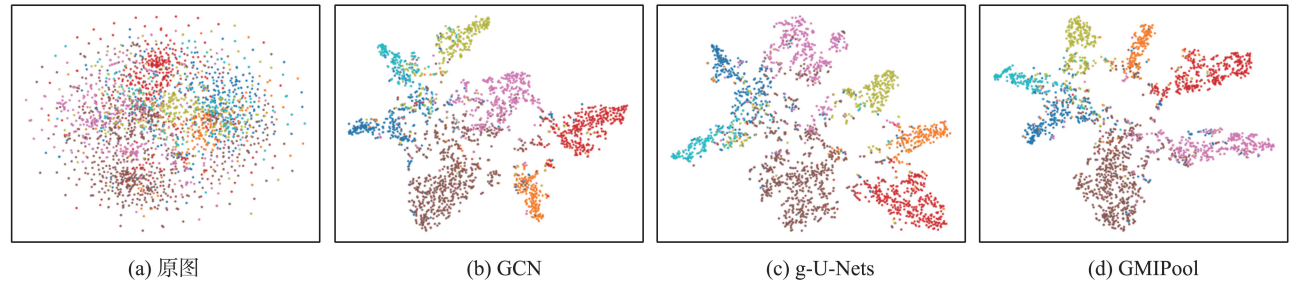


图 3 不同图神经网络方法生成的节点嵌入的可视化示意图

Fig.3 Visualization of node embeddings generated by different graph neural network methods

3.5 池化层数分析

为验证 GMIPool 在提高 GNNs 表达力方面的能力,本实验针对不同池化层数的 GMIPool 分析,实验中记录数据集 Cora、Citeseer 和 Pubmed 上不同池化层数的节点分类准确率。由于池化率大小是灵活多变的,因此在实验中除第一个池化层以外,其他池化层的池化率均设为 0.8。如表 3 所示,当深度小于 4 时,随着池化层数的增加,模型在各数据集上的分类精度逐渐提高,说明 GMIPool 有效地减少特征冗余,保留更加紧凑的结构信息和特征差异且能够在一定程度上缓解 GNNs 中常见的拉普拉斯过平滑问题^[25]。此外,在池化层数较小的情况下,GMIPool 在各数据集上的分类准确率比普通的 GCN 高,表明即使在浅层网络中,GMIPool 依然能有效提取图的分层信息。

表 3 不同层数 GMIPool 在各数据集上的分类准确率

Table 3 Classification accuracy of GMIPool with different layers on various datasets

单位:%

池化层数	Cora	Citeseer	Pubmed
1	83.1±0.5	70.7±0.6	79.6±0.4
2	83.4±0.3	72.2±0.4	80.4±0.4
3	84.5±0.7	72.7±0.6	80.6±0.6
4	84.9±0.5	74.0±0.2	80.9±0.7
5	84.1±0.3	71.3±0.3	80.4±0.3

3.6 参数敏感性分析

本节分析 GMIPool 对于超参数的敏感性。本实验分析 GMIPool 中参数 α 、 λ 以及隐藏层特征维度大小 d 的对分类准确率的影响。其中,参数 α 用于调节模型对池化损失和分类损失的重视程度, α 会随着训练次数增

加逐步降低,因此本实验测试 α 的初始值对于GMIPool的影响, λ 用于调节重构过程对于粗图的影响, d 直接影响模型的表达和学习能力,较大的特征维度能够提高模型的学习能力,但是存在过拟合的风险。

实验中, $\alpha \in [0, 5]$, $\lambda \in [0, 1]$,如图4所示,当 α 和 λ 为0时,GMIPool在各数据集上的表现最差,侧面证实GMIPool池化方法和粗图构建是有效的。当 $\alpha \in [0, 3]$ 时,GMIPool在各数据集上表现较好,对于参数 λ ,在数据集Pubmed, λ 变化波动略大,但在数据集Cora和Citeseer上较为稳定,这可能是由于数据集Pubmed中节点个数较多,实验中使用较大的池化率,导致GMIPool对于参数 λ 较为敏感。最后,实验测试不同特征维度 d 对分类性能的影响, $d \in [16, 32, 48, 64, 128]$,当 $d=32$ 时,GMIPool在各数据集上分类准确率达到最高。

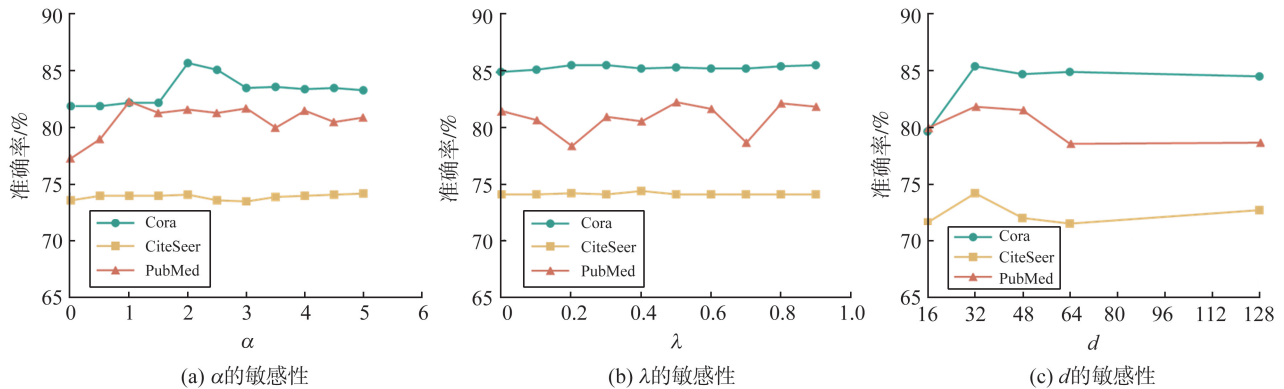


图4 GMIPool各参数敏感性分析示意图

Fig.4 Illustration of sensitivity analysis of each parameter of GMIPool

4 结语

本文提出一种图池化算子GMIPool,GMIPool通过评估节点和其支撑图之间的图互信息识别图中的关键子结构,使得GNNs能够学习图中的层次结构。本文在节点分类的5个数据集上进行广泛实验,证明了GMIPool的有效性。

局部互信息的权重可以视作其对于全局互信息的贡献程度,不同的局部互信息对于全局互信息的贡献程度应该是不一样的,即局部MI之间可能存在某种程度上的偏序关系。在支撑图构建方面,未来将尝试其它支撑图构建方式,利用偏序关系探索更加合理的互信息估计方式以及更加灵活的支撑图构建方法并在其它任务上验证GMIPool的有效性。

参考文献:

- [1] LI Zewen, LIU Fan, YANG Wenjie, et al. A survey of convolutional neural networks: analysis, applications, and prospects[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 33(12):6999-7019.
- [2] WU Zonghan, PAN Shirui, CHEN Fengwen, et al. A comprehensive survey on graph neural networks[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 32(1):4-24.
- [3] DWIVEDI V P, JOSHI C K, LUU A T, et al. Benchmarking graph neural networks[J]. Journal of Machine Learning Research, 2023, 24(43):1-48.
- [4] LIU Chuang, ZHAN Yibing, WU Jia, et al. Graph pooling for graph neural networks: progress, challenges, and opportunities [C]//Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence. Macao: IJCAI, 2023:6712-6722.
- [5] BIANCHI F M, LACHI V. The expressive power of pooling in graph neural networks[J]. Advances in Neural Information Processing Systems, 2023, 36:71603-71618.
- [6] YU Shujian, GIRALDO L G S, PRÍNCIPE J C. Information-theoretic methods in deep neural networks: recent advances and emerging opportunities[C]//Proceedings of the Thirty International Joint Conference on Artificial Intelligence. Montreal: IJCAI, 2021:4669-4678.
- [7] HJELM R D, FEDOROV A, LAVOIE-MARCHILDON S, et al. Learning deep representations by mutual information estimation and maximization[C]//International Conference on Learning Representations. New Orleans, Louisiana: OpenReview.net, 2019.
- [8] VELICKOVIC P, FEDUS W, HAMILTON W L, et al. Deep graph infomax [C] // International Conference on Learning

- Representations. New Orleans, Louisiana; OpenReview.net, 2019.
- [9] PENG Zhen, HUANG Wenbing, LUO Minnan, et al. Graph representation learning via graphical mutual information maximization[C]//Proceedings of The Web Conference. New York: ACM, 2020:259-270.
- [10] PANG Yunsheng, ZHAO Yunxiang, LI Dongsheng. Graph pooling via coarsened graph infomax[C]//Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 2021:2177-2181.
- [11] VINYALS O, BENGIO S, KUDLUR M. Order matters: sequence to sequence for sets[C]//International Conference on Learning Representations. San Juan: AAAI, 2016.
- [12] ZHANG Muhan, CUI Zhicheng, NEUMANN M, et al. An end-to-end deep learning architecture for graph classification[C]//Proceedings of the AAAI Conference on Artificial Intelligence. New Orleans, Louisiana: AAAI, 2018, 32(1):4438-4445.
- [13] YING Zitao, YOU Jiaxuan, CHRISTOPHER M, et al. 2018. Hierarchical graph representation learning with differentiable pooling[C]//Proceedings of the 32nd International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Incoporation, 2018:4805-4815.
- [14] YUAN Had, JI Shuiwang. Structpool: structured graph pooling via conditional random fields[C]//Proceedings of the 8th International Conference on Learning Representations. Addis Ababa: IEEE, 2020.
- [15] MA Yao, WANG Suhang, TANG Liliang, et al. Graph convolutional networks with eigenpooling[C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM, 2019:723-731.
- [16] GAO Hongyang, JI Shuiwang. Graph U-nets[C]//International Conference on Machine Learning. California: PMLR, 2019:2083-2092.
- [17] LEE J, LEE I, KANG J. Self-attention graph pooling[C]//International Conference on Machine Learning. California: PMLR, 2019:3734-3743.
- [18] GAO Xing, DAI Wenrui, LI Chenglin, et al. Ipool-information-based pooling in hierarchical graph neural networks[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 33(9):5032-5044.
- [19] ZHANG Zhen, BU Jiajun, MARTIN E, et al. Hierarchical multi-view graph pooling with structure learning[J]. IEEE Transactions on Knowledge and Data Engineering, 2021, 35(1):545-559.
- [20] LI Maosen, CHEN Siheng, ZHANG Ya, et al. Graph cross networks with vertex infomax pooling[J]. Advances in Neural Information Processing Systems. Red Hook: Curran Associates Incoporation, 2020, 33:14093-14105.
- [21] KIPF T N, WELING M. Semi-supervised classification with graph convolutional networks[C]//International Conference on Learning Representations. Toulon: OpenReview.net, 2017.
- [22] VELIČKOVIĆ P, CUCURULL G, CASANOVA A, et al. Graph attention networks[C]//International Conference on Learning Representations. Vancouver: OpenReview.net, 2018.
- [23] HAMILTON W L, YING Zitao, LESKOVEC J. Inductive representation learning on large graphs[C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach: Curran Associates Incoporation, 2017:1025-1035.
- [24] ZHONG Zhiqiang, LI Chengte, PANG Jun. Multi-grained semantics-aware graph neural networks[J]. IEEE Transactions on Knowledge and Data Engineering, 2022, 35(7):7251-7262.
- [25] ZHANG Xu, XU Yonghui, HE Wei, et al. A comprehensive review of the oversmoothing in graph neural networks[C]//CCF Conference on Computer Supported Cooperative Work and Social Computing. Singapore: Springer, 2023:451-465.

(编辑:陈丽萍)