



基于偏序关系的多视图多粒度图表示学习框架

肖添龙, 徐计, 王国胤

引用本文:

肖添龙, 徐计, 王国胤. 基于偏序关系的多视图多粒度图表示学习框架[J]. *智能系统学报*, 2025, 20(1): 243-254.

XIAO Tianlong, XU Ji, WANG Guoyin. A multi-view and multi-granularity graph representation learning framework based on partial order relations[J]. *CAAI Transactions on Intelligent Systems*, 2025, 20(1): 243-254.

在线阅读 View online: <https://dx.doi.org/10.11992/tis.202406010>

您可能感兴趣的其他文章

基于级联宽度学习的多模态材质识别

Cascade broad learning for multi-modal material recognition

智能系统学报. 2020, 15(4): 787-794 <https://dx.doi.org/10.11992/tis.201908021>

基于Capsule网络的甲骨文构件识别方法

Recognition of Oracle Radical based on the Capsule network

智能系统学报. 2020, 15(2): 243-254 <https://dx.doi.org/10.11992/tis.201904069>

图神经网络推荐研究进展

Research advances in graph neural network recommendation

智能系统学报. 2020, 15(1): 14-24 <https://dx.doi.org/10.11992/tis.201908034>

基于多粒度结构的网络表示学习

Network representation learning based on multi-granularity structure

智能系统学报. 2019, 14(6): 1233-1242 <https://dx.doi.org/10.11992/tis.201905045>

基于改进卷积神经网络的多标记分类算法

A multi-label classification algorithm based on an improved convolutional neural network

智能系统学报. 2019, 14(3): 566-574 <https://dx.doi.org/10.11992/tis.201804056>

高斯核函数卷积神经网络跟踪算法

Convolutional neural network tracking algorithm accelerated by Gaussian kernel function

智能系统学报. 2018, 13(3): 388-394 <https://dx.doi.org/10.11992/tis.201612040>

DOI: 10.11992/tis.202406010

基于偏序关系的多视图多粒度图表示学习框架

肖添龙^{1,2}, 徐计¹, 王国胤³

(1. 贵州大学 省部共建公共大数据国家重点实验室, 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; 3. 重庆邮电大学 计算智能重庆市重点实验室, 重庆 400065)

摘要: 图池化作为图神经网络中重要的组件, 在获取图的多粒度信息的过程中扮演了重要角色。而当前的图池化操作均以平等地位看待数据点, 普遍未考虑利用邻域内数据之间的偏序关系, 从而造成图结构信息破坏。针对此问题, 本文提出一种基于偏序关系的多视图多粒度图表示学习框架 (multi-view and multi-granularity graph representation learning based on partial order relationships, MVMGr-PO), 它通过从节点特征视图、图结构视图以及全局视图对节点进行综合评分, 进而基于节点之间的偏序关系进行下采样操作。相比于其他图表示学习方法, MVMGr-PO 可以有效地提取多粒度图结构信息, 从而可以更全面地表征图的内在结构和属性。此外, MVMGr-PO 可以集成多种图神经网络架构, 包括 GCN(graph convolutional network)、GAT(graph attention network) 以及 GraphSAGE(graph sample and aggregate) 等。通过在 6 个数据集上进行实验评估, 与现有基线模型相比, MVMGr-PO 在分类准确率上有明显提升。

关键词: 图神经网络; 图池化; 多粒度; 偏序关系; 节点分类任务; 图表示学习; 半监督学习; 图嵌入

中图分类号: TP391 **文献标志码:** A **文章编号:** 1673-4785(2025)01-0243-12

中文引用格式: 肖添龙, 徐计, 王国胤. 基于偏序关系的多视图多粒度图表示学习框架 [J]. 智能系统学报, 2025, 20(1): 243-254.

英文引用格式: XIAO Tianlong, XU Ji, WANG Guoyin. A multi-view and multi-granularity graph representation learning framework based on partial order relations[J]. CAAI transactions on intelligent systems, 2025, 20(1): 243-254.

A multi-view and multi-granularity graph representation learning framework based on partial order relations

XIAO Tianlong^{1,2}, XU Ji¹, WANG Guoyin³

(1. State Key Laboratory of Public Big Data, Guizhou University, Guiyang 550025, China; 2. College of Computer Science and Technology, Guizhou University, Guiyang 550025, China; 3. Chongqing Key Laboratory of Computational Intelligence, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: Graph pooling, as a crucial component of graph neural networks (GNNs), plays a vital role in capturing multi-granularity information of graphs. However, current graph pooling operations typically treat data points equally, often neglecting the partial order relationships among data within neighborhoods, which leads to the disruption of graph structural information. To address this issue, we propose a novel framework for multi-view and multi-granularity graph representation learning based on partial order relationships, named MVMGr-PO. This framework comprehensively scores nodes from the perspectives of node feature view, graph structure view, and global view, and then performs down-sampling operations based on the partial order relationships among nodes. Compared with other graph representation learning methods, MVMGr-PO effectively extracts multi-granularity graph structural information, thus providing a more comprehensive representation of the intrinsic structure and attributes of the graph. Additionally, MVMGr-PO can integrate various graph neural network (GNN) architectures, including graph convolutional network (GCN), graph attention network (GAT), and graph sample and aggregate (GraphSAGE). Experimental evaluations on six datasets demonstrate that compared with existing baseline models, MVMGr-PO significantly improves classification accuracy.

Keywords: graph neural networks; graph pooling; multi-granularity; partial order relationships; node classification task; graph representation learning; semi-supervised learning; graph embedding

收稿日期: 2024-06-07.

基金项目: 国家自然科学基金项目 (62366008, 61966005, 62221005).

通信作者: 徐计. E-mail: jixu@gzu.edu.cn.

©《智能系统学报》编辑部版权所有

过去的主流深度学习框架主要处理欧几里得空间的数据, 如计算机视觉^[1-2]、自然语言处理^[3]等, 对其进行特征学习和预测。现如今, 非欧几

里得空间存在大量数据,例如社交网络、分子结构等,其内部通过大量的节点和链接进行表示。而对于图这种非结构化数据而言,其内部的拓扑结构以及非欧几里得性质难以使用欧几里得空间模型进行表示。因此,为有效地学习和处理图数据,需要探索新的深度学习方法,使其能够适应非欧几里得空间中的图结构,从而拓宽深度学习的应用领域,如社区网络分析^[4]、蛋白质结构预测^[5]、推荐系统^[6]等。图表示学习是这一领域中的一个重要方向,它旨在解决如何在非欧几里得空间中捕捉和表达图数据。

近年来,图神经网络(graph neural networks, GNNs)^[7]发展迅速,其主要思想是通过逐层的信息传递和聚合来捕捉图结构的信息。它采用表示学习的思想,将节点表示为一个向量,该向量捕捉了节点的特征及其邻域的拓扑结构。通过逐层的信息传递与聚合,GNNs能够捕捉从局部到全局的广泛信息,实现对图数据的全面分析。其学习到的节点表示被广泛用于各种图分析任务,如节点分类^[8-10]、连接预测^[11]以及图分类^[12]等。

单层的 GNNs 聚合了节点的一阶邻域信息,随着层数的增加,GNNs 可以捕捉到更广泛的邻域信息,但是也可能产生过平滑问题,特别是在节点分类任务中,过平滑问题将导致模型性能大幅下降。此外,随着层数增加,单个节点嵌入向量的计算量将呈指数级增加。而在大数据时代的背景下,图中的节点以及边的数量非常庞大,这使得 GNNs 的原始实现在大规模数据上变得不可行。对此,研究者在模型架构、采样策略、聚合技术等方面进行设计,在一定程度上缓解了这种局面,例如基于改进采样策略的 GraphSAGE (graph sample and aggregate)^[13]和 Cluster-GCN (cluster generative adversarial network)^[14]等模型。然而,这些方法总是受限于计算资源和模型复杂度,对于大规模图数据的处理仍然存在挑战。

在图分析中,了解社区结构的层次性对于理解图的组织特性至关重要。社区可被视为图的层次结构,其中节点形成了具有内在联系的子图。然而,对于大规模的图数据,处理其内部复杂的层次结构具有挑战性。为了应对这一挑战,分层池化方法成为一种有效的技术手段。分层池化可以根据粗化的策略大致分为节点聚合池化^[15-16]以及节点丢弃池化^[17-19]。节点聚合池化是将细粒度图的节点进行聚合从而形成粗粒度图的节点,具体来说主要分 2 步——获取聚类分配矩阵、使用聚类分配矩阵获取粗粒度图,而节点丢弃池化

主要分 3 步——节点分数评估、节点选择、使用选择的节点形成粗粒度图。尽管现有的分层池化方法为处理大规模图数据提供了有效的解决方案,但它们也存在一些缺点。例如,节点聚合池化在下采样过程中,相似节点的聚合可能会改变原始图的拓扑结构,导致图的整体拓扑结构发生变化。这种拓扑结构的改变可能会影响到后续图分析任务的准确性和可靠性。而节点丢弃池化在下采样过程中,使用固定的池化率会导致对所有节点的处理方式一致,无法充分考虑到节点的重要性差异,从而可能导致重要节点被过度丢弃或者不重要的节点被保留,最终影响了图的表征质量。

为了解决上述问题,本文提出一种基于偏序关系的多视图多粒度图表示学习框架(a multi-view and multi-granularity graph representation learning framework based on partial order relations, MVM-Gr-PO)。它从节点特征视图、图结构视图以及全局视图分别对节点进行评分,再使用注意力机制计算权重对 3 种视图的评分进行融合,进一步提高模型对节点重要性评估的准确性和鲁棒性。随后,基于节点之间的偏序关系自适应地确定关键节点而无需指定池化率。这使得 MVMGr-PO 同时继承节点聚合以及节点丢失池化的优点,同时避免了图结构破坏、过度丢弃以及不重要的节点被保留的问题。通过这种自适应地确定关键节点的方法,模型能够更加灵活地适应不同图数据的特点,从而提高模型在图分析任务中的性能和泛化能力。

1 相关工作

最近,多粒度图表示学习^[20]的研究领域经历了迅速的发展,其目标是探索如何更好地捕捉图数据的多尺度信息,从而提高图表示学习任务的性能和可解释性。分层池化方法作为一种强大的工具,已经在多粒度图表示学习领域展现出巨大的潜力。该方法旨在通过层次化的聚合和抽象图中的节点信息,来捕捉图数据的多尺度特性。分层池化可以根据粒化的方式分为节点聚合池化和节点丢弃池化。节点聚合池化和节点丢弃池化作为分层池化方法的两个重要分支,在多粒度图表示学习领域中扮演着至关重要的角色。

1) 节点聚合池化。节点聚合池化将池化看作是节点聚类分配问题,将细粒度层的节点进行聚合生成粗粒度层的节点。现有的各种节点聚合池化方法的主要区别在于聚类分配矩阵的获取方式不同。DIFFPOOL (differentiable pooling)^[15]使用

GNNs 生成可微分的聚类分配矩阵 S , 实现将节点软分配到各个类簇。此外 DIFFPOOL 引入辅助链接预测任务以及熵正则化使其更容易训练。而 MgPool^[21] 获取聚类分配矩阵 S 使用多层感知机 (multi-layer perceptron, MLP), 同样将节点软分配到各个类簇。MILE (multi-level embedding)^[22] 采用结构等价匹配 (structural equivalence matching, SEM) 和归一化重边匹配 (normalized heavy edge matching, NHEM) 对图进行逐层粒化, 在最粗糙的图上使用现有的嵌入技术进行嵌入计算。MIVS-Pool(maximal independent vertex set pooling)^[23] 使用图卷积网络 (GCN)^[24] 对每个节点进行评分, 然后选择一个最大独立集的节点作为粗粒度节点, 并将其余节点分配给粗粒度节点以形成簇, 从而获得聚类分配矩阵。SEP (structural entropy guided graph hierarchical pooling)^[25] 通过结构熵最小化, 可以将图的层次结构解码为相应的编码树, 将噪声或随机变化引起的扰动最小化, 从而根据编码树获取聚类分配矩阵。而后续处理基本上都是通过使用聚类分配矩阵计算出粗粒度层的特征矩阵和邻接矩阵, 从而完成图粗化。

2) 节点丢弃池化。与节点聚合池化不同的是, 节点丢弃池化不会将节点聚合成更粗粒度的节点, 而是选择性地丢弃一部分节点, 剩余节点构成粗粒度图的节点。现有的各种节点丢弃池化方法的主要区别在于节点评分的方式不同。Graph U-Nets^[17] 根据节点在可训练投影向量上的标量投影值自适应地选择一些节点, 以形成一个粗粒度图。具体来说, 计算节点的特征向量在投影向量 p 上的标量投影值, 得到的投影值表示该节点在投影向量 p 上的信息保留程度。SAGPool (self-attention graph pooling)^[26] 使用 GCN 计算节点的评分, 再通过指定池化率 k 来保留分数排名为前 $[kN]$ 的节点作为粗粒度层中的节点。ASAP (adaptive structure aware pooling)^[19] 为每个节点考虑 h 跳邻域内的邻居作为中心节点的类簇, 通过注意力机制计算节点在每个类簇中的成员关系, 再使用局部极值卷积 (local extrema convolution, LEConv)^[19] 计算每个类簇的适应度分数, 最后根据适应度分数选择排名前 $[kN]$ 个类簇中心节点作为粗粒度层的节点。MVPool (multi-view graph pooling)^[18] 利用图结构和节点特征信息进行全面和客观的节点评估, 从多个角度得到节点的评分并使用注意力机制进行融合, 最终选择一定比率的重要节点作为粗粒度图的节点。此外, MVPool 考虑到丢弃池化会导致图结构的破坏, 因此提出使用稀疏注意

力机制从诱导子图中重新学习一个优化后的图结构。CovPooling (covariance pooling)^[27] 通过计算节点嵌入的协方差矩阵, 并利用节点之间的相关系数生成节点得分, 选取前 $[kN]$ 的节点生成图的层次表示。

2 具体方法

2.1 符号定义

给定一个图 $G = (V, A, F)$, 其中 V 表示节点集合; A 是图的邻接矩阵, 代表节点之间的连接关系, $A_{ij} = 1$ 表示节点 v_i 和节点 v_j 之间存在边的连接, 反之 $A_{ij} = 0$; 节点的特征矩阵为 $F \in \mathbb{R}^{n \times f}$, n 表示图中节点个数, f 为图的原始特征的维度。

由于图池化的影响, 每一层图的结构会变化, 为区分各个粒层的图, 使用 $G_k = (V_k, A_k, X_k)$ 来表示第 k 个粒层的图, 相应的 $V_k, A_k \in \mathbb{R}^{n_k \times n_k}$, $X_k \in \mathbb{R}^{n_k \times d}$, $Z_k \in \mathbb{R}^{n_k \times d}$ 分别表示第 k 层粒层图的节点集、邻接矩阵、输入表示以及输出表示。特别的, 对于原始图来说, n_0 表示原始图中节点个数, 那么输入表示为 $X_0 = F \in \mathbb{R}^{n_0 \times f}$ 。

2.2 基本定义

定义 1 偏序关系^[28] 假设 S 是一个集合, \leq 是集合 S 上的二元关系, 若 \leq 满足以下条件:

$$\begin{cases} \forall a \in S, a \leq a \\ \forall a, b \in S, (a \leq b \wedge b \leq a) \Rightarrow a = b \\ \forall a, b, c \in S, (a \leq b \wedge b \leq c) \Rightarrow a \leq c \end{cases} \quad (1)$$

则称 \leq 为集合 S 上的偏序关系。

定义 2 偏序集^[29] 若在集合 S 上给定一个偏序关系 \leq , 则称集合 S 按偏序关系 \leq 构成一个偏序集合, 记作 (S, \leq) 。

2.3 下采样

在卷积神经网络 (convolutional neural networks, CNNs)^[30] 中, 池化层扮演着至关重要的角色。下采样也被称为池化, 旨在降低特征图的大小并扩大其感受野, 使其能够对输入的空间变化更具鲁棒性。而图池化可以看作是卷积神经网络中池化概念在图神经网络中的延伸, 其发挥着与 CNNs 池化层相似的作用。与 CNNs 池化层不同的是 GNNs 的池化层处理的是非欧几里得空间的图数据, 因此其实现方式更加复杂。

在本小节中, 介绍 MVMGr-PO 的下采样 (图池化) 操作。与传统的图池化方法不同, 该操作不需要预设池化率, 这使得它更具灵活性和通用性, 能够适用于不同规模和结构的图数据。此外, 它采用多视图评估策略来全面而准确地衡量每个节点的重要性。这种策略不仅考虑节点的局

部特征,还融入了图的整体结构信息,从而能够更加精准地确定哪些节点在池化过程中应该被保留。

首先将图中的每个节点 v_i 视为一个类簇的中心点,每个类簇包含距离该中心点固定半径 h 跳内的邻居,即 $c_h(v_i) = \mathcal{N}_h(v_i)$ 。特别的,当 $h=1$ 时, $c(v_i)$ 即为节点 v_i 的所有邻居节点构成的集合 $\mathcal{N}(v_i)$ 。在这种设定下,每个节点都会形成一个局部的类簇,这些类簇覆盖了图的不同区域,并捕捉了图的局部结构特征。MVMGr-PO 将图池化视为一个类簇选择问题,即从众多的簇中选择出最具代表性的簇来代表整个细粒度图。为实现这一目标,本文设计一种基于多视图的类簇重要性评分机制。具体而言,MVMGr-PO 从节点特征视图、图结构视图以及全局视图对每个类簇进行综合评分。

1) 节点特征视图。这一视图基于节点的特征信息来评估其重要性。通过提取节点的输入表示 \mathbf{X}_k ,使用 MLP 来判断哪些类簇中心节点在特征空间上具有代表性或独特性。节点的输入表示 \mathbf{X}_k 使得 MVMGr-PO 能够保留对图整体表示具有贡献的类簇中心节点:

$$s_f = \text{sigmoid}(\text{MLP}(\mathbf{X}_k)) \quad (2)$$

式中: $s_f \in \mathbb{R}^n$ 表示在节点特征视图下计算类簇中心节点的重要性得分, $\mathbf{X}_k \in \mathbb{R}^{n \times d}$ 表示第 k 个粒层图的输入特征。

2) 图结构视图。这一视图主要关注每个类簇在图中的位置以及类簇之间的连接关系。MVMGr-PO 使用类簇的大小 $|c_h(v_i)|$ 来反映以节点 v_i 为中心的类簇在图结构中重要性。特别的,当类簇半径 $h=1$ 时, $|c_h(v_i)|$ 即为节点 v_i 的度。在真实的网络中,节点的度分布大多数服从幂律分布^[31],幂律分布意味着少数节点拥有非常高的度,而大多数节点的度相对较低。类似的,中心节点 v_i 所代表类簇的大小 $|c_h(v_i)|$ 也服从幂律分布,这导致评分过于偏向于那些 $|c_h(v_i)|$ 大的类簇,而忽视了其他类簇的重要性。为防止幂律分布导致的偏差,一种常见的做法是对 $|c_h(v_i)|$ 进行某种形式的转换或调整,以使得评分符合特定的分析需求。使用对数函数 (log) 是一种常见的方法,对数函数可以将较大的数值压缩到较小的范围内,同时保持数值之间的相对差异。然而,直接使用对数可能会导致所有类簇的分数变得非常接近,从而失去了区分度。因此,MVMGr-PO 采用平方根函数 (sqrt) 来处理幂律分布所带来的问题。平方根函数能够在一定程度上减小 $|c_h(v_i)|$ 数值之间的巨大差异,同时能够保持一定的区分度,使得那些具有较大 $|c_h(v_i)|$ 的类簇在评分中仍然能够脱颖而

出,但又不至于过于偏向于它们。此外,引入一些可学习参数来进一步调整平方根函数,以使其更加灵活地适应不同的幂律分布和评分需求。参数 α 用于控制评分的整体缩放,通过调整 α 的值,可以控制评分的大小范围,以便更好地适应特定的评分系统或可视化需求。参数 β 是一个平移因子,用于控制分数的平移,进而可以在不改变评分相对差异的情况下,整体提高或降低所有类簇的评分。这有助于将评分调整到特定的范围或满足特定的分析需求。具体计算公式为

$$s_s = \text{sigmoid}(\alpha \times \text{sqrt}(|c_h(v)|) + \beta) \quad (3)$$

式中: $s_s \in \mathbb{R}^n$ 表示在图结构视图下计算类簇中心节点的重要性得分, $|c_h(v)| \in \mathbb{R}^n$ 表示各中心节点所代表的类簇大小。

3) 全局视图。这一视图旨在结合前两个视图的信息,从全局角度评估节点的重要性。通过综合考虑节点的局部结构、特征信息,可以得到更加全面和准确的节点重要性评分。在综合考虑节点的局部结构和特征信息时,GCN 和 MLP 是两种常用的方法。GCN 能够有效地捕捉节点之间的关系和局部结构,而 MLP 则可以对节点的特征进行非线性变换和组合。将它们结合起来可以提高节点重要性评分的准确性和全面性。具体计算公式为

$$s_{sf} = \text{sigmoid}(\text{MLP}(\mathbf{X}_k) + \text{GCN}(\mathbf{B}_k, \mathbf{X}_k)) \quad (4)$$

式中: $s_{sf} \in \mathbb{R}^n$ 表示在全局视图下计算类簇中心节点的重要性分数; $\mathbf{B}_k = \sum_{i=1}^h (\mathbf{A}_k)^i$,即将类簇中心节点与其 h 跳范围内的所有邻居节点进行连接,从而形成一个更广泛的连接图结构; $\mathbf{X}_k \in \mathbb{R}^{n \times d}$ 表示第 k 个粒层图的输入表示; sigmoid 函数的作用是将分数映射到 $[0, 1]$,使得各个视图的分数可以在相同的尺度上进行比较。

在综合考虑节点特征视图、图结构视图和全局视图的信息后,MVMGr-PO 使用加权注意力机制动态地将 3 种分数结合,以得到最终的节点重要性评分。

具体的,定义注意力权重 α_i 用于第 i 个视图,其计算方式为

$$\alpha_i = \frac{\exp(\sigma(\varphi w_i + b_i))}{\sum_{j=1}^3 \exp(\sigma(\varphi w_j + b_j))} \quad (5)$$

式中: $\varphi = [s_f, s_s, s_{sf}] \in \mathbb{R}^{n \times 3}$ 表示 3 种视图下分数的拼接, $w_i \in \mathbb{R}^3$ 和 $b_i \in \mathbb{R}$ 是视图 i 的可学习参数。最后,将注意力权重与各个视图的重要性分数相乘,得到最终的类簇中心节点的重要性评分 ϕ ,具

体计算公式为

$$\phi = \sum_{i=1}^3 \alpha_i \cdot z_i \quad (6)$$

式中 z_i 是视图 i 下的节点重要性得分。

接下来, 使用偏序关系来确定图中的关键节点。引入偏序关系使得 MVMGr-PO 能够对节点的重要性进行比较和排序, 进而识别出在整个图结构中扮演关键角色的节点。现考虑一个偏序集 $G = (V, >)$, 其中 V 表示节点集合, $>$ 表示“大于”关系。现为每个类簇中心节点 $v_i \in V$ 都赋予一个对应的评分 ϕ_i 。为了从细粒度图的节点集合中选出关键节点, 现定义集合 M , 表示所有“节点的评分高于其所有相邻节点的评分”的节点, 即

$$M = \{v_i \mid \phi_i > \phi_j, \forall v_j \in N(v_i)\} \quad (7)$$

式中 $N(v_i)$ 表示节点 v_i 的邻居集合。集合 M 中的节点在其局部邻域内具有最高的评分, 因此将其作为核心节点构建粗粒度层。这些节点在网络结构中拥有显著的地位, 代表着其邻域节点。除了将集合 M 中的节点视为重要节点外, 还需保留不属于集合 M 中节点邻域的节点, 以维持图结构的完整性。现定义集合 N 为

$$N = V - \bigcup_{v_i \in M} N(v_i) \quad (8)$$

式中 \cup 表示并集运算。因此, 经过粗化处理, 得到的节点总数为 $|M| + |N|$ 个。粗粒度图的构造示意如图 1 所示。

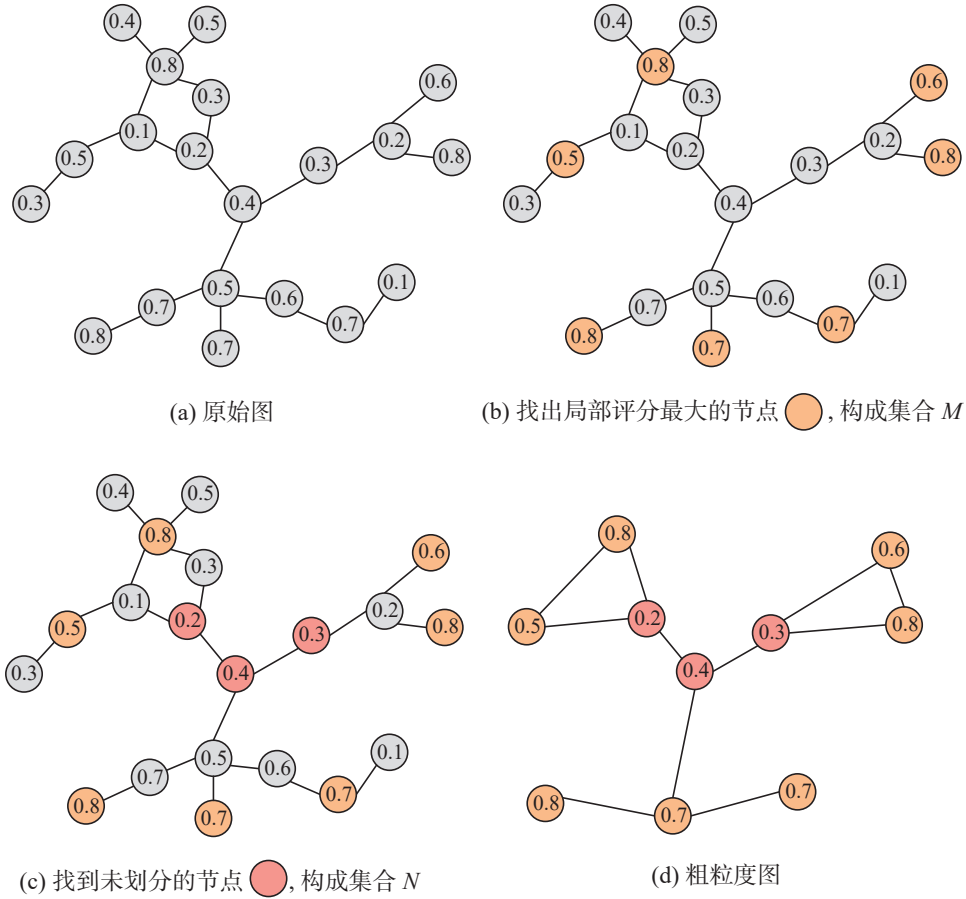


图 1 粗粒度图的构造示意

Fig. 1 Illustration of the construction of a coarse-grained graph

延续节点聚合池化的思想, 构造分配矩阵 $S_{k,k+1} \in \mathbb{R}^{n^k \times (|M| + |N|)}$ 。其中, 第 k 个粒度层的节点个数为 n^k , 第 $k+1$ 个粒度层的节点个数为 $n^{k+1} = |M| + |N|$ 。 $S_{k,k+1}$ 的每一行代表细粒度图的每一个节点, 而每一列代表粗粒度图中的节点。由于集合 M 和 N 中的节点在粗粒度层中被保留, 因此对于 $\forall v_i \in M \cup N$, 有 $S_{k,k+1}[i, j] = 1$, 其中 j 表示节点 v_i 在粗粒度图中的对应节点索引。此外, 对于 $\forall v_j \in M$ 且

$v_i \in N(v_j)$, 有 $S_{k,k+1}[i, j] = 1$, 其中 j 表示节点 v_j 在粗粒度图中的对应节点索引, v_i 为节点 v_j 在细粒度层中的邻居。此外, 对于 $S_{k,k+1}$ 的其他位置填充为 0。需要注意的是, 该方法本质上是一个软分配, 即一个细粒度节点可能同时被分配到多个粗粒度节点中, 这反映了在粒度转换过程中节点可能具有的多重归属或多重关系。

通过给定的分配矩阵 $S_{k,k+1}$ 可以将细粒度图中

的节点聚合为粗粒度图中的节点。给定细粒度图的邻接矩阵 $\mathbf{A}_k \in \mathbb{R}^{n^k \times n^k}$, 使用分配矩阵 $\mathbf{S}_{k,k+1} \in \mathbb{R}^{n^k \times n^{k+1}}$ 将其转换为粗粒度层图的邻接矩阵 $\mathbf{A}_{k+1} \in \mathbb{R}^{n^{k+1} \times n^{k+1}}$, 具体操作为

$$\mathbf{A}_{k+1} = \mathbf{S}_{k,k+1}^T \mathbf{A}_k \mathbf{S}_{k,k+1} \quad (9)$$

这使得在细粒度层中, 如果两个超节点共享任何公共节点, 则它们在粗粒度层中将保持连接。在构建粗粒度图时, 通常希望每个粗粒度节点能够保留细粒度图中的一些重要特征信息。这些特征信息可以用来描述节点在整个图结构中的重要性、性质或其他相关属性。一种常见的方法是基于聚合操作, 通过对细粒度图中节点的特征进行聚合来构造粗粒度图节点的特征, 即使用 $\mathbf{X}_{k+1} = \mathbf{S}_{k,k+1}^T \mathbf{Z}_k \in \mathbb{R}^{n^{k+1} \times d}$ 。然而, 该方式通过对细粒度图中邻居节点的特征进行聚合可能会导致关键节点的特征信息被模糊, 从而丢失自己的原始信息, 使其无法充分反映整个图结构的性质。因此, 本文采用如下方法来解决这一问题:

$$\begin{aligned} \tilde{\mathbf{Z}}_k &= \mathbf{Z}_k[i_{dx}, :] \in \mathbb{R}^{n^{k+1} \times d} \\ \tilde{\phi} &= \phi[i_{dx}, :] \in \mathbb{R}^{n^{k+1} \times 1} \\ \mathbf{X}_{k+1} &= \tilde{\mathbf{Z}}_k \odot (\tilde{\phi} \mathbf{1}_d^T) \in \mathbb{R}^{n^{k+1} \times d} \end{aligned} \quad (10)$$

式中: i_{dx} 表示粗粒度图中节点在细粒度图中对应索引; $\mathbf{Z}_k[i_{dx}, :]$ 和 $\phi[i_{dx}, :]$ 分别用来提取出粗粒度节点在细粒度图中的输出表示 $\tilde{\mathbf{Z}}_k$ 以及节点评分 $\tilde{\phi}$; $\mathbf{1}_d^T$ 表示一个长度为 d 的元素全为 1 的行向量, 则 $\tilde{\phi} \mathbf{1}_d^T \in \mathbb{R}^{n^{k+1} \times d}$; \odot 表示矩阵逐元素乘积。将粗粒度节点的输出表示与评分进行乘积运算, 旨在实现两者信息的融合。通过这种方式, 评分能够作为输出表示的权重, 影响其在后续计算和分析中的地位。更重要的是, 这使得评分可以通过反向传播算法进行更新。

2.4 上采样

与下采样相对应, 上采样是图神经网络中的一个重要模块, 它用于恢复细粒度图的图结构和节点嵌入表示。上采样的目标是利用粗粒度图的信息来获取细粒度图的信息, 从而实现细粒度图的重建。由于在池化过程中, 通过保存各粒度层的邻接矩阵 \mathbf{A}_k 已经获得各个粒度层的图结构, 目前还缺少各个粒度层节点的嵌入表示 \mathbf{e}_k 。接下来, 将讨论如何来填补这一缺失。

首先, 本文使用分配矩阵 $\mathbf{S}_{k,k+1}$ 将粗粒度图的嵌入表示 \mathbf{e}_{k+1} 进行投影从而得到细粒度图的嵌入表示 \mathbf{e}_k^p :

$$\mathbf{e}_k^p = \mathbf{S}_{k,k+1} \mathbf{e}_{k+1} \quad (11)$$

式中 $\mathbf{e}_{k+1} \in \mathbb{R}^{n_{k+1} \times d}$ 是粗粒度图的嵌入表示。对于最粗粒度图来说 $\mathbf{e}_m = \mathbf{X}_m$, m 是池化层数。由于直接

将粗粒度节点的嵌入投影到细粒度节点上, 因此称 $\mathbf{e}_k^p \in \mathbb{R}^{n^k \times d}$ 为第 k 个粒度层的投影嵌入。使用分配矩阵 $\mathbf{S}_{k,k+1}$ 来指导投影过程, 为的是确保投影后的细粒度节点表示能够保持与粗粒度节点的关联性和一致性。

由于在下采样过程中, 集合 M 中节点 v_i 的邻居都被 v_i 所代表从而变成一个粗粒度节点, 这种简单的投影将导致节点 v_i 的邻居节点共享相同的嵌入。此外, 在上文中提到, 一个细粒度节点可能会被分配到多个粗粒度节点中, 这导致多个粗粒度节点的嵌入信息将聚合在同一个细粒度节点上, 从而导致细粒度节点的嵌入表示平滑化。基于上面提到的这两个原因, 需要在投影之后通过引入额外的机制对节点的嵌入进行提纯和优化。在这里, 本文采用残差连接并使用 GNNs 对 $\mathbf{e}_k^p \oplus \mathbf{Z}_k$ 进行优化, 具体来说:

$$\mathbf{e}_k = \text{GNNs}(\mathbf{A}_k, \mathbf{e}_k^p \oplus \mathbf{Z}_k) \quad (12)$$

式中: $\mathbf{e}_k \in \mathbb{R}^{n^k \times d}$ 为第 k 粒度层的最终嵌入表示, $\mathbf{Z}_k \in \mathbb{R}^{n^k \times d}$ 为下采样过程中的输出表示, \oplus 表示矩阵逐元素相加。

2.5 模型框架

节点表示学习目标是找到一个映射函数 $f: V \mapsto \mathbb{R}^d$, 即将图中的每个节点 v 映射为一个 d 维实向量 $f(v)$, 且 $d \ll n$, 其中 n 为图中节点个数。节点分类的任务是根据节点的表示向量来获取该节点的类别 y 。因此需要一个目标标签矩阵 $\mathbf{Y} \in \mathbb{R}^{n \times c}$, 其中 c 表示类别数。若节点 v_i 属于类别 j 则 $Y_{ij} = 1$ 否则 $Y_{ij} = 0$ 。MVMGr-PO 进行节点表示学习如算法 1 所述。

算法 1 MVMGr-PO 算法

输入 输入图 $G = (V, \mathbf{A}, \mathbf{F})$, 类簇半径 h , 下采样层数 m

输出 节点的嵌入表示 $\mathbf{e}_0 \in \mathbb{R}^{n^0 \times d}$

//下采样阶段

- 1) For $k = 0, 1, 2, \dots, m-1$ do;
- 2) 获得输出表示 $\mathbf{Z}_k = \text{GNN}(\mathbf{X}_k, \mathbf{A}_k)$;
- 3) 计算 s_f, s_s 以及 s_{sf} ; //式 (2)(3)(4)
- 4) 使用注意力机制计算节点分数 ϕ ; //式 (5)(6)
- 5) 获得粗粒度图节点集 $V_{k+1} = M \cup N$; //式 (7)(8)
- 6) 构造分配矩阵 $\mathbf{S}_{k,k+1}$;
- 7) 获得粗粒度图的邻接矩阵 \mathbf{A}_{k+1} ; //式 (9)
- 8) 获得粗粒度图的节点特征表示 \mathbf{X}_{k+1} ; //式 (10)
- 9) End For
- 10) 获得最粗粒度图的嵌入表示 $\mathbf{e}_m = \mathbf{X}_m$;

//上采样阶段

- 11) For $k = m-1, m-2, \dots, 0$ do;

- 12) 得到投影嵌入 ε_k^p ; //式 (11)
- 13) 残差连接后使用 GNNs 优化投影嵌入; //式 (12)
- 14) End For;
- 15) Return ε_0 。

MVMGr-PO 进行节点表示学习如算法 1 所述。首先, 将原始图数据及其相应的节点特征输入 GNNs 中进行嵌入表示, 以获取初步的节点表示。接下来, 对节点进行下采样(池化)操作, 将图数据从细粒度逐步转换为粗粒度, 并通过逐层应用 GNNs 进行学习以提取更高层次的特征。在

经过一系列的 GNNs 和下采样操作后, 获得最粗粒度图的嵌入表示。随后, 利用式 (11) 将这些嵌入投影到细粒度图上。同时, 在投影嵌入到细粒度图上的过程中, 利用残差连接将下采用过程中记录的各层输出表示 Z_k 加到投影嵌入 ε_k^p 上。这种残差连接的机制使模型能够保留更多的细节信息, 并且加速模型收敛过程。进一步地, 通过再次应用 GNNs 对 $\varepsilon_k^p \oplus Z_k$ 进行优化。这个过程重复进行, 直到最终得到原始图的最终嵌入表示。节点表示学习框架示意如图 2 所示。

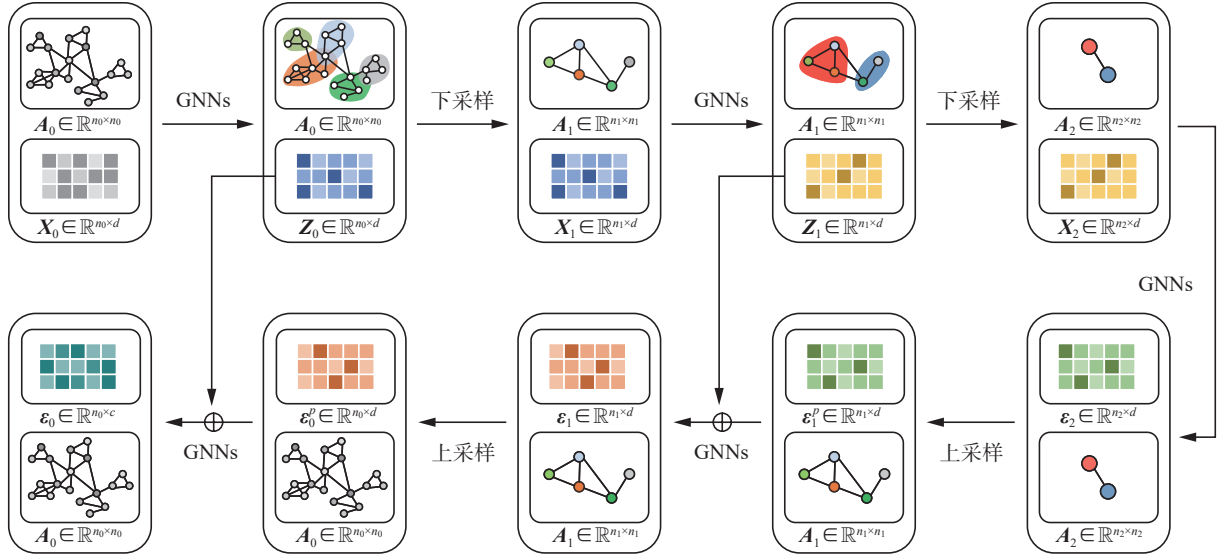


图 2 节点表示学习框架示意

Fig. 2 Illustration of the node representation learning framework

通过这种节点表示学习框架, 模型能够在保留原始图结构信息的同时, 有效地捕捉多个粒度图中节点的特征, 并在分类任务中取得优秀的性能表现。

3 实验部分

本节中, 将 MVMGr-PO 与最先进的节点分类模型进行比较。接下来将介绍数据集、基线方法、实验设置以及实验结果的分析讨论。

3.1 数据集

本实验采用半监督的直推式学习 (transductive), 模型能够访问所有节点的属性信息以及整个图结构。为了全面比较本实验方法与最先进方法的性能, 本文选取了 6 个广泛使用的数据集作为实验数据集。这些数据集包括 3 个引文数据集 (Citeseer、Pubmed 和 Cora), 这些数据集中的节点和边分别表示不同主题的研究论文和它们之间的引用关系; 2 个合著关系数据集 (CS 和 Physics), 在这些数据集中节点代表作者, 边表示作者之间的共同作者关系, 即两位作者共同发表的一篇或

多篇论文; 以及 1 个亚马逊商品共现网络数据集 Photo, 其中节点表示商品, 边表示在同一交易中购买的商品。

由于采用半监督学习, 在 Cora、Citeseer 以及 Pubmed 数据集中, 对于每个类别随机选择 20 个节点作为训练集, 500 个节点作为验证集以及 1000 个节点作为测试集。对于 CS、Physics 和 Photo 数据集, 本实验对每个类别随机选择 20 个节点进行训练, 30 个节点作为验证集, 其余节点作为测试集。其统计数据如表 1 所示。

表 1 数据集统计数据
Table 1 Information statistic of datasets

数据集	节点数目	边数目	特征维度	类别数	数据集划分
Cora	2 708	10 556	1 433	7	140/500/1 000
Citeseer	3 327	9 104	3 703	6	120/500/1 000
Pubmed	19 717	88 648	500	3	60/500/1 000
CS	18 333	163 788	6 805	15	300/450/17 583
Physics	34 493	495 924	8 415	5	100/150/34 243
Photo	7 650	238 162	745	8	160/240/7 250

3.2 基线方法

本实验将节点分类基线方法大致分为 4 种:

1) 传统节点表示学习方法。在这一类中, 本实验选取 MLP、DeepWalk^[32] 以及 Node2vec^[33] 作为基准方法。MLP 通过将节点特征映射至一个表示向量, 进而用于节点分类任务。DeepWalk 将随机游走产生的节点序列视作自然语言处理中的句子, 其中每个节点的邻居节点在游走过程中扮演了类似于 Word2Vec 模型中上下文窗口的角色。通过这种方法, DeepWalk 成功地将词嵌入技术应用于图数据, 实现了节点的有效嵌入。Node2vec 是 DeepWalk 的扩展, 它通过调整随机游走策略, 在探索图结构时能够平衡局部和全局信息, 进一步提升了节点嵌入的质量。

2) 基于图神经网络的节点嵌入方法。这类方法是传统的图神经网络, 为节点分类提供了高效解决方案。GCN^[24] 通过多层图卷积分聚合邻居信息, 生成富含结构信息的嵌入。GAT (graph attention network)^[34] 引入注意力机制, 使模型能够自适应地学习不同邻居节点的重要性, 并据此进行信息聚合。

3) 基于节点聚合池化的分层表示模型。这类算法通过聚合节点及其邻居的信息, 形成图的层次化表示, 从而捕捉图数据的内在结构和特征。AdamGNN^[16] 将图池化视为自网络选择问题, 通过计算每个自网络的分数来选择关键节点进行池化, 并根据自网络中节点的隶属关系构造分配矩阵, 从而将节点进行聚合以形成粗粒度节点。

4) 基于节点丢弃池化的分层表示模型。这类算法通过设置池化率, 选择并丢弃图中的节点从而实现图的层次化表示。Graph U-Nets^[17] 根据节点在可训练投影向量上的标量投影值选择一些节点以形成粗粒度图。MVPool^[18] 通过结合不同视图的信息来生成稳健的节点排名, 再通过指定池化率来实现图池化。

3.3 实验设置

本实验采用 PyTorch Geometric^[35] 实现 MVMGr-PO, 具体代码已上传至 github (<https://github.com/>

[codefly-xtl/MVMGr-PO](#))。在模型训练过程中, 采用 Adam^[36] 优化器进行训练。在进行每次训练时, 在验证集上进行验证, 并保存在验证集上准确率最高的模型。最后, 使用保存的模型在测试集上进行一次测试。这种训练流程确保了模型在验证集上的表现是整个训练过程中最优秀的模型。而通过在独立的测试集上进行测试, 可以评估模型的泛化能力和在未见过的数据上的性能。此外, 本实验对模型进行 10 次评估, 并计算平均准确率以及标准差, 以确保评估结果的稳定性和可靠性。这种多次评估的方法有助于减少由于数据随机性或模型初始化的差异而引起的评估偏差。

在节点分类任务中, 本实验对不同的数据集有不同的参数设置, 具体的参数设置如表 2 所示。此外, 对于基线模型 GCN 和 GAT, 由于文献^[35] 和本实验有着相同的数据划分规则, 本实验直接引用 PyTorch Geometric 的 benchmark^[35] 中的性能结果。对于 AdamGNN、Graph U-Nets 以及 MVPool 模型, 本实验遵循原论文中的实验参数设置进行对比实验。

表 2 模型参数
Table 2 Model parameters

数据集	学习率	权重 衰减	隐藏层 单元数	丢弃率	池化 层数	类簇 半径	迭代 次数
Cora	0.01	0.01	256	0.8	2	2	20
Citeseer	0.001	0.1	256	0.7	1	2	50
Pubmed	0.01	0.001	256	0.8	1	1	50
CS	0.01	0.01	256	0.7	2	1	50
Physics	0.01	0.01	256	0.8	1	1	20
Photo	0.05	0.001	256	0.5	3	1	100

3.4 实验结果

为了验证 MVMGr-PO 的整体性能, 本实验将 MVMGr-PO 与基线模型进行了对比。表 3 给出了在各个公开数据集上的分类准确度, 其中粗体表示该数据集上的最佳性能, 而性能排名第 2 的数据则使用下划线进行显示。

表 3 在各个公开数据集上的分类准确度
Table 3 Classification accuracy on each public dataset

模型	Cora	Citeseer	Pubmed	CS	Physics	Photo
MLP	53.4±5.0	56.2±2.9	68.2±4.6	86.4±1.3	88.0±1.4	78.7±1.0
DeepWalk	67.2±0.8	41.2±2.0	64.1±2.3	78.2±1.5	76.6±1.6	73.1±1.8
Node2Vec	67.0±1.5	43.0±2.0	63.7±2.9	77.9±1.3	75.7±1.7	72.5±2.1
GCN	79.4±1.9	68.1±1.7	77.4±2.4	88.0±1.2	91.5±2.6	87.2±0.1
GAT	81.0±1.4	69.2±1.9	78.3±2.3	90.3±0.7	91.6±1.0	87.3±0.1

%

续表 3

模型	Cora	Citeseer	Pubmed	CS	Physics	Photo
GraphSAGE	78.9±1.7	66.6±2.3	75.0±2.9	90.5±0.5	90.0±1.4	<u>91.3±0.1</u>
AdamGNN	83.1±0.9	<u>72.8±0.5</u>	<u>80.6±0.6</u>	91.5±0.1	93.7±0.2	90.2±0.2
MVPool	83.6±1.2	72.6±1.1	80.0±1.8	92.0±0.5	94.0±0.6	88.8±0.4
Graph U-Nets	<u>83.9±0.4</u>	72.2±0.3	80.6±0.7	<u>92.2±0.1</u>	<u>94.2±0.1</u>	88.9±0.5
MVMGr-PO	85.1±0.2	75.2±0.3	83.8±0.2	92.5±0.1	94.7±0.1	92.2±0.1
相对提升	1.4	3.2	3.9	0.3	0.5	0.9

Deepwalk 和 Node2vec 方法虽然利用图结构信息来获取节点嵌入, 但却忽视了节点特征信息的重要性。GNNs 通过将节点特征和图结构信息编码到一个统一的表示空间中, 实现显著的性能提升。具体到 GCN、GAT 以及 GraphSAGE, 它们相比传统的 MLP 和基于随机游走的嵌入技术有所进步, 但仍未充分整合和利用图中的多粒度信息。进一步的, 像 AdamGNN、MVPool 和 Graph U-Nets 虽然强调了多粒度信息的整合, 但在实际应用中暴露出一些局限性。例如, AdamGNN 采用节点聚合池化思想对图进行粗化, 但其对重要节点的选择策略过于简单, 可能无法有效识别和保留图中最具代表性的节点。而 MVPool 和 Graph U-Nets 采用节点丢弃池化策略, 通过设定池化率来进行池化, 而不合适的池化率可能导致不重要的节点被保留, 或者重要节点被过度丢弃, 会造成大量孤立的节点从而破坏图的结构完整性。MVMGr-PO 通过构建不同的视图来评估每个节点的重要性, 并利用节点的评分以及节点间的偏序关系来识别和保留关键节点从而实现有效的图粗化。这种自适应的池化策略允许模型在不同粒度层次上捕捉和保留图的关键信息。

实验证明, MVMGr-PO 在多个数据集上的表现均优于现有方法, 充分证明了其有效性。具体而言, 与不具备分层表示学习机制的 GCN 相比, MVMGr-PO 在各个数据集上相对提高了 7.1%、10.4%、8.2%、5.1%、3.4% 以及 5.7%, 这验证了多粒度节点表示学习的必要性。相较于各个数据集中性能排名第 2 的基线模型, MVMGr-PO 在各个数据集上的相对提升分别达到了 1.4%、3.2%、3.9%、0.3%、0.5% 以及 0.9%。这些结果不仅验证

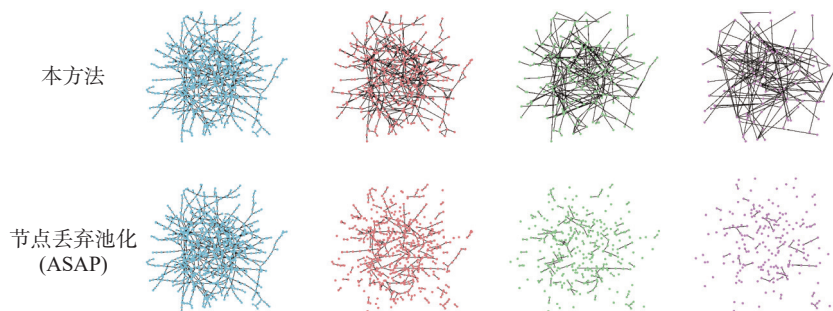
了 MVMGr-PO 的有效性, 也强调了在图表示学习中考虑节点间偏序关系的重要性。

3.5 模型分析

3.5.1 下采样策略可视化分析

在本小节, 通过可视化技术比较 MVMGr-PO 方法与传统的节点丢弃池化和节点聚合池化方法的区别。为此, 本实验使用 Watts 模型^[32]构建了一个小世界图, 该图包含 1000 个节点, 每个节点与其在环形拓扑结构中的 3 个最近邻居节点相连, 并设置重连概率为 0.3。为了揭示图的底层拓扑结构, 本实验采用力导向布局 (force-directed layout) 进行可视化。在下采样 (池化) 操作中, MVMGr-PO 无需指定池化率, 而节点丢弃池化和节点聚合池化的池化率均设定为 0.6。

图 3 给出了在相同的图上应用 3 种不同下采样策略后得到的结果, 每种方法均执行了 3 次池化操作。MVMGr-PO 方法通过多视图对节点进行综合评价, 并基于节点间的偏序关系来识别图中的重要节点。这种策略旨在保留结构上更为关键的节点, 从而在减少节点数量的同时维持图的基本框架。从结果可见, 即便经过 3 次池化, 图的骨架结构依然得到很好的保留。相较之下, 以 ASAP^[19]代表的节点丢弃池化方法在每次迭代后均导致大量边的丢失, 图的结构信息显著减少, 图的连通性和信息传递能力可能遭受严重影响。节点聚合池化方法则是通过聚合细粒度节点形成粗粒度节点从而实现图池化。本文以 MgPool^[21]为例进行展示, 可视化结果揭示了过度聚合引发的问题, 包括引入许多非原始存在的边, 这些新边扭曲了图的原始结构信息。



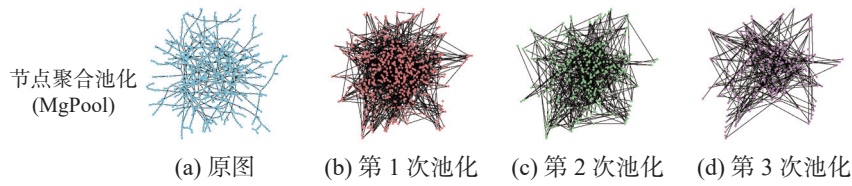


图 3 不同池化方法的可视化

Fig. 3 Visualization of different pooling methods

通过可视化比较,可以清晰地观察到,相较于节点丢弃池化和节点聚合池化, MVMGr-PO 在保留图的基本结构方面展现出了显著的优势。这种保留原始图结构的能力对于后续的图分析任务至关重要,因为它确保了图的核心信息在池化过程中得以完整保留,为后续分析提供了坚实的基础。

3.5.2 多视图评估策略的影响分析

在粗粒度节点选择过程中,评估节点分数的策略对实验结果有着关键影响。本研究通过对比不同的节点评分方法,旨在验证 MVMGr-PO 中多

视图评估策略的有效性。

实验设计包括为图中每个节点随机分配评分,作为基准进行性能比较。此外,本实验引入其他几种不同的节点评分策略作为对比。具体而言,参照 Graph U-Nets 的方法,通过计算节点特征向量在一个可训练投影向量上的标量投影值来确定节点分数。进一步,借鉴 SAGPool 的策略,使用图神经网络 (GNNs) 计算节点分数。最后,采用 ASAP 中的局部极值卷积 (local extrema convolution, LEConv) 模块计算节点分数,以全面评估各种方法的有效性。实验结果如图 4 所示。

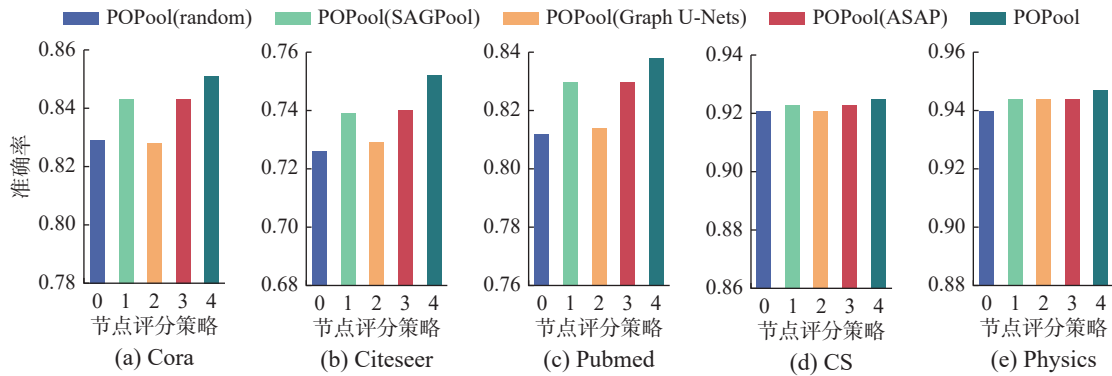


图 4 不同的节点分数评估策略对性能的影响

Fig. 4 Impact of different node scoring evaluation strategies on performance

实验数据表明,尽管评分策略的选择对模型准确性的影响在一定程度上是有限的,但不同的策略在不同数据集上的表现仍展现出了一定的差异性。

相较之下,采用了 Graph U-Nets 的评分策略相比于基线方法略有提升,而使用 SAGPool 和 ASAP 的评分策略在 Cora、Citeseer 以及 Pubmed 数据集上表现较为突出。相对于各种评分策略, MVMGr-PO 模型在性能上取得较好的性能。这表明,通过采用多视图评估策略, MVMGr-PO 能够更准确地选择粗粒度节点。

4 结束语

本研究针对现有图表示学习方法中未充分考虑节点间的偏序关系,导致图结构信息和关键节点特征的丢失问题,提出一种基于偏序关系的多

视图多粒度图表示学习框架。该框架综合考虑了节点特征视图、图结构视图以及全局视图,通过多视图分析节点的重要性,并依据节点间的偏序关系执行下采样操作,有效保持了图结构的完整性,使模型能够在不同层次上捕捉图数据的特征。在上采样阶段,采用先投影后优化的策略,从而避免了两个问题:一是隶属于同一类簇的细粒度节点的特征平滑化;二是多个粗粒度节点嵌入表示聚合在同一细粒度节点上,导致该节点的嵌入表示平滑化。此外,使用残差连接将下采样与上采样阶段进行连接,能够加速模型收敛过程。实验结果表明,本模型具有一定的有效性。

本研究虽在性能上取得了一定的提升,但仍有进一步的完善空间。在未来的工作中,一方面考虑使用图结构学习对下采样之后的图结构进行优化;另一方面考虑引入多任务学习机制,以提

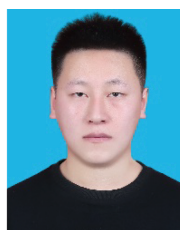
高模型的泛化能力。

参考文献:

- [1] SCHMARJE L, SANTAROSSA M, SCHRÖDER S M, et al. A survey on semiself and unsupervised learning for image classification[J]. *IEEE access*, 2021, 9: 82146–82168.
- [2] 陈洛轩, 林成创, 郑招良, 等. Transformer 在计算机视觉场景下的研究综述[J]. *计算机科学*, 2023, 50(12): 130–147.
CHEN Luoxuan, LIN Chengchuang, ZHENG Zhaoliang, et al. Review of transformer in computer vision[J]. *Computer science*, 2023, 50(12): 130–147.
- [3] LAURIOLA I, LAVELLI A, AIOLLI F. An introduction to deep learning in natural language processing: models, techniques, and tools[J]. *Neurocomputing*, 2022, 470: 443–456.
- [4] BLONDEL V D, GUILLAUME J L, LAMBIOTTE R, et al. Fast unfolding of communities in large networks[J]. *Journal of statistical mechanics: theory and experiment*, 2008, 2008(10): 10008.
- [5] SAKAI M, NAGAYASU K, SHIBUI N, et al. Prediction of pharmacological activities from chemical structures with graph convolutional neural networks[J]. *Scientific reports*, 2021, 11(1): 525.
- [6] GAO Chen, WANG Xiang, HE Xiangnan, et al. Graph neural networks for recommender system[C]//Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining. Virtual Event: ACM, 2022: 1623–1625.
- [7] 吴国栋, 查志康, 涂立静, 等. 图神经网络推荐研究进展[J]. *智能系统学报*, 2020, 15(1): 14–24.
WU Guodong, ZHA Zhikang, TU Lijing, et al. Research advances in graph neural network recommendation[J]. *CAAI transactions on intelligent systems*, 2020, 15(1): 14–24.
- [8] 刘彦北, 马夕然, 王雯. 可信的图神经网络节点分类方法[J]. *天津工业大学学报*, 2024, 43(1): 82–88.
LIU Yanbei, MA Xiran, WANG Wen. Node classification method based on trusted graph neural network[J]. *Journal of Tiangong university*, 2024, 43(1): 82–88.
- [9] 东昱晓, 柯庆, 吴斌. 基于节点相似性的链接预测[J]. *计算机科学*, 2011, 38(7): 162–164, 199.
DONG Yuxiao, KE Qing, WU Bin. Link prediction based on node similarity[J]. *Computer science*, 2011, 38(7): 162–164, 199.
- [10] 王兆慧, 沈华伟, 曹婧, 等. 图分类研究综述[J]. *软件学报*, 2022, 33(1): 171–192.
WANG Zhaohui, SHEN Huawei, CAO Qi, et al. Survey on graph classification[J]. *Journal of software*, 2022, 33(1): 171–192.
- [11] HAMILTON W L, YING R, LESKOVEC J. Inductive representation learning on large graphs[EB/OL]. (2017–06–07)[2024–06–07]. <https://arxiv.org/abs/1706.02216v4>.
- [12] CHIANG Weilin, LIU Xuanqing, SI Si, et al. Cluster-GCN: an efficient algorithm for training deep and large graph convolutional networks[C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Anchorage: ACM, 2019: 257–266.
- [13] YING Z, YOU Jiaxuan, MORRIS C, et al. Hierarchical graph representation learning with differentiable pooling [J]. *Advances in neural information processing systems*, 2018, 31: 1–11.
- [14] ZHONG Zhiqiang, LI Chengte, PANG Jun. Multi-grained semantics-aware graph neural networks[J]. *IEEE transactions on knowledge and data engineering*, 2023, 35(7): 7251–7262.
- [15] GAO H, JI S. Graph U-Nets[J]. *IEEE transactions on pattern analysis and machine intelligence*, 2022, 44(9): 4948–4960.
- [16] ZHANG Zhen, BU Jiajun, ESTER M, et al. Hierarchical multi-view graph pooling with structure learning[J]. *IEEE transactions on knowledge and data engineering*, 2023, 35(1): 545–559.
- [17] RANJAN E, SANYAL S, TALUKDAR P. ASAP: adaptive structure aware pooling for learning hierarchical graph representations[J]. *Proceedings of the AAAI conference on artificial intelligence*, 2020, 34(4): 5470–5477.
- [18] 张蕾, 钱峰, 赵姝, 等. 基于多粒度结构的网络表示学习 [J]. *智能系统学报*, 2019, 14(6): 1233–1242.
ZHANG Lei, QIAN Feng, ZHAO Shu, et al. Network representation learning based on multi-granularity structure[J]. *CAAI transactions on intelligent systems*, 2019, 14(6): 1233–1242.
- [19] SUN Haichao, WANG Guoyin, LIU Qun. Graph neural networks with multi-granularity pooling[C]//2022 IEEE 8th International Conference on Cloud Computing and Intelligent Systems. Chengdu: IEEE, 2022: 114–118.
- [20] LIANG Jiongqian, GURUKAR S, PARTHASARATHY S. MILE: a multi-level framework for scalable graph embedding[J]. *Proceedings of the international AAAI conference on web and social media*, 2021, 15: 361–372.
- [21] STANOVIC S, GAÜZÈRE B, BRUN L. Maximal independent vertex set applied to Graph pooling[M]//Lecture Notes in Computer Science. Cham: Springer International Publishing, 2022: 11–21.

- [22] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks[EB/OL]. (2016-09-09)[2024-06-07]. <https://arxiv.org/abs/1609.02907v4>.
- [23] WU Junran, CHEN Xueyan, XU Ke, et al. Structural entropy guided graph hierarchical pooling[C]//International conference on machine learning. New York: PMLR, 2022: 24017-24030.
- [24] LEE J, LEE I, KANG J. Self-attention graph pooling[C]//International conference on machine learning. New York: PMLR, 2019: 3734-3743.
- [25] JIANG Jianjian, LEI Fangyuan, DAI Qingyun, et al. Graph pooling in graph neural networks with node feature correlation[C]//Proceedings of the 3rd International Conference on Data Science and Information Technology. Xia'men: ACM, 2020: 105-110.
- [26] KIM Y. Convolutional neural networks for sentence classification[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. Stroudsburg: Association for Computational Linguistics, 2014: 1746-1751.
- [27] PEROZZI B, AL-RFOU R, SKIENA S. DeepWalk: on-line learning of social representations[C]//Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. New York: ACM, 2014: 701-710.
- [28] GROVER A, LESKOVEC J. node2vec: scalable feature learning for networks[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco: ACM, 2016: 855-864.
- [29] Veličković P, CUCURULL G, CASANOVA A, et al. Graph attention networks[EB/OL]. (2017-10-30)[2024-06-07]. <https://arxiv.org/abs/1710.10903>.
- [30] FEY M, LENSSEN J E. Fast graph representation learning with PyTorch geometric[EB/OL]. (2019-03-06)[2024-06-07]. <https://arxiv.org/abs/1903.02428v3>.
- [31] KINGMA D P, BA J, HAMMAD M M. Adam: a method for stochastic optimization[EB/OL]. (2014-12-22)[2024-06-07]. <https://arxiv.org/abs/1412.6980v9>.
- [32] WATTS D J, STROGATZ S H. Collective dynamics of 'small-world' networks[J]. *Nature*, 1998, 393(6684): 440-442.
- [33] GROVER A, LESKOVEC J. Node2vec: scalable feature-learning for networks[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2016: 855-864.
- [34] VELICKOVIC P, CUCURULL G, CASANOVA A, et al. Graph attention networks[J]. *Stat*, 2017, 1050(20): 39-41.
- [35] FEI M, LENSSEN J E. Fast graph representation learning with PyTorch geometric[C]//ICLR Workshop on Representation Learning on Graphs and Manifolds. Appleton: ICLR, 2019: 1-9.
- [36] KINGMA D P, Ba J. Adam: a method for stochastic optimization[EB/OL]. (2022-12-22)[2024-06-07]. <https://arxiv.org/abs/1412.6980>.

作者简介:



肖添龙, 硕士研究生, 主要研究方向为图神经网络、粒计算和机器学习。E-mail: gs.tlxiao22@gzu.edu.cn。



徐计, 特聘教授, 博士, 主要研究方向为数据挖掘、粒计算和机器学习。主持国家自然科学基金项目 2 项, 主持大型互联网企业横向项目 2 项, 出版学术专著 1 部, 发表学术论文 20 余篇。E-mail: jixu@gzu.edu.cn。



王国胤, 教授, 博士生导师, 国家级人才, 重庆师范大学校长。主要研究方向为粗糙集、粒计算、数据挖掘、认知计算、大数据、人工智能。曾任国际粗糙集学会 (IRSS) 理事长, 现任中国人工智能学会 (CAAI) 副理事长、中国计算机学会 (CCF) 理事、重庆市人工智能学会 (CQAAI) 理事长, IRSS/CAAI/CCF 会士。获国内外发明专利授权 20 余项, 出版学术专著和教材 20 多部 (含编著), 发表学术论文 300 余篇, 论著被他人引用 10 000 多次。E-mail: wanggy@cqupt.edu.cn。