计算机研究与发展
Journal of Computer Research and Development
DOI: 10.7544/issn1000-1239.202440142
61(8): 1993−2007, 2024

HSEGRL: 一种分层可自解释的图表示学习模型

李 平[1,2] 宋舒寒[1,2] 张 园[1,2] 曹华伟[1,3] 叶笑春[1] 唐志敏[1]
[1](处理器芯片全国重点实验室(中国科学院计算技术研究所) 北京 100190)
[2](中国科学院大学 北京 100049)
[3](中国科学院大学南京学院 南京 211135)
(liping20b@ict.ac.cn)

HSEGRL: A Hierarchical Self-Explainable Graph Representation Learning Model

Li Ping[1,2], Song Shuhan[1,2], Zhang Yuan[1,2], Cao Huawei[1,3], Ye Xiaochun[1], and Tang Zhimin[1]
[1] (*State Key Lab of Processors* (*Institute of Computing Technology*, *Chinese Academy of Sciences*), *Beijing* 100190)
[2] (*University of Chinese Academy of Sciences*, *Beijing* 100049)
[3] (*University of Chinese Academy of Sciences*, *Nanjing* 211135)

Abstract In recent years, with the extensive application of graph neural network (GNN) technology in the fields such as social network, information, chemistry and biology, the interpretability of GNN has attracted widespread attention. However, prevailing explanation methods fail to capture the hierarchical explanation information, and these hierarchical information has not been fully utilized to improve the classification accuracy of graph tasks. To address this issue, we propose a hierarchical self-explanation graph representation learning model called HSEGRL (hierarchical self-explanation graph representation learning). This model, by discovering hierarchical information in the graph structure, predicts graph classifications while outputting hierarchical self-explanation results. Specifically, we design the basic unit for extracting hierarchical information—interpreters. These interpreters consist of an encoder that extracts node features, a pooling layer that perceives hierarchical explanation-aware subgraphs, and a decoder that refines higher-order explanation information. We refine the pooling mechanism with an explanation-aware strategy, enabling the hierarchical selection of subgraphs based on the evaluation of the model's topology and feature importance, thereby facilitating hierarchical self-explanation in conjunction with graph classification. HSEGRL is a functionally comprehensive and transferable self-explanatory graph representation learning framework that can hierarchically consider the model's topological information and node feature information. We conduct extensive experiments on datasets from molecular, protein, and social network, and demonstrate that HSEGRL surpasses existing advanced self-explanatory graph neural network models and graph neural network models in terms of graph classification performance. Furthermore, the visualization of layered explanation outcomes substantiates the credibility of our proposed explanation methodology.

Key words graph representation learning; graph neural networks; self-explainable model; graph topology; message passing mechanism

摘 要 近年来,随着图神经网络(graph neural network, GNN)技术在社交、信息、化学、生物等领域的广

收稿日期: 2024-03-15; **修回日期**: 2024-05-20
基金项目: 国家重点研发计划项目(2023YFB4502305); 北京市自然科学基金项目(4232036); CAS 青年创新促进会项目
This work was supported by the National Key Research and Development Program of China (2023YFB4502305), the Beijing Natural Science Foundation (4232036), and the CAS Project for Youth Innovation Promotion Association.
通信作者: 曹华伟(caohuawei@ict.ac.cn)

泛应用，GNN 可解释性也受到广泛的关注. 然而，现有的解释方法无法捕获层次化的解释信息，同时，这些层次信息未能被充分利用以提升图分类任务的准确率. 基于这一问题，提出了一种层次化自解释的图表示学习（hierarchical self-explanation graph representation learning, HSEGRL）模型，该模型通过发现图结构中的层次信息进行图分类预测的同时，输出层次化的模型自解释结果. 具体而言，针对图层次信息的发现设计了提取信息的基本单元——解释子，该解释子由提取节点特征的编码器获取层次化解释感知子图的池化层和抽取高阶解释信息的解码器组成. 其中，为了准确提取层次化的解释子图，针对该模型的池化操作进行了解释感知优化设计，该设计通过评估模型的拓扑及特征重要性，层次化地筛选解释子图，实现分层自解释的同时完成图分类任务.HSEGRL 是一个功能完备且便于迁移的图表示学习自解释模型，可以层次化综合考虑模型的拓扑信息与节点特征信息. 在模型有效性验证层面，分别在分子、蛋白质和社交数据集上进行大量实验，实验结果表明所提模型在图分类任务中的分类准确率高于已有的先进的 GNN 自解释模型和 GNN 模型，并通过可视化分层解释结果的信息证明了该解释方法可信.

关键词 图表示学习；图神经网络；可自解释模型；图拓扑；消息传递机制

中图法分类号 TP183

随着互联网技术的迅速发展，以节点和边表征关系信息的图结构，由于其具有强大的关系表达能力，使其在关联关系分析中变得越来越重要. 图表示学习（graph represention learning, GRL）是指将图结构或图节点映射为向量表示的任务，其主要目标是将节点映射为向量表示的同时尽可能多地保留图的拓扑信息. 图神经网络（graph neural network, GNN）被设计用于处理基于图结构的数据，广泛应用于知识图谱[1]、推荐系统[2]和社交网络[3]等各个领域，在聚类、节点分类、链接预测和图分类等任务中都能取得较好的性能和精度[4]. 随着 GNN 的发展，GNN 用于处理图表示学习的问题的研究也受到广泛关注.

GNN 模型采用消息传递机制将节点和图表示成向量的形式. 具体而言，每个节点通过聚合其邻居节点的信息来获得自己的表示[5-6]，这种方法使得 GNN 能够有效地捕捉图数据中编码的局部和全局结构信息. 然而，当涉及以清晰易懂的方式分析和预测图数据时，GNN 仍然存在一些局限性，如人们很难理解 GNN 是如何做出决策的. 随着图数据规模的不断增长，图数据的处理难度也呈指数级增加，为 GNN 模型的透明度的提升带来了极大挑战. GNN 模型的解释在增强模型透明度方面发挥着至关重要的作用，这有助于在实际应用中优先考虑公平、隐私和安全等问题[7]. 因此，越来越多的工作将优化目标聚焦于如何提升 GNN 的可解释性.

目前，已有的研究工作提出了一些方法来增强 GNN 的可解释性，主要分为事后解释（post-hoc explanation）和自解释（self-explanation）两类. 其中，事后解释是针对已经完成了训练的目标 GNN 模型，根据训练模型的输入和输出来解释结果. 一些早期的 GNN 解释性工作，如 GNNExplainer[8] 和 PGExplainer[9] 等，都是基于事后解释的方法. GNNExplainer 通过最大化 GNN 预测和可能的子图结构分布之间的互信息来解释现有的 GNN 模型. 然而，GNNExplainer 只能预测单个实例的解释信息，难以对多个实例同时进行解释，因此缺乏对于 GNN 模型解释的全局视角. 对此，PGExplainer 通过一个参数化的深度神经网络从训练好的 GNN 模型中学习解释的生成过程，实现了对多个实例同时进行解释.

然而，正如前文所述，文献 [8-9] 方法都是针对事后解释设计的，即通过分析模型输入输出的关系预测模型做出决策的原因，因此极易受输入输出数据的偏差而影响解释结果. 此外，事后可解释 GNN 模型仍然是个黑盒模型，依旧不能获取模型内部学习到的规则及决策过程. 基于上述问题，可自解释 GNN 模型被设计出来. 可自解释 GNN 模型在进行模型训练的同时，捕获模型解释相关的节点特征或拓扑结构，使模型在进行分类和预测任务的同时，对于自身做出的决策进行解释. 典型的可自解释 GNN 工作包括 SEGNN, CAL, Orphicx 等模型. SEGNN 模型[10]提出了一种全新 GNN 自解释框架，通过为每个未标记的节点找到距离最近的标记节点，来提供可解释的节点分类结果. CAL 模型[11]针对图分类任务，在 GNN 模型中引入因果注意力机制，来减轻无关特征对于解释结果的混淆. Orphicx[12] 通过构建一个特定的生成模型，优化模型的目标函数，激励该生成模型产生因果的、可信的解释.

从上述描述可以发现，事后解释模型所需的参

数较少，所需的计算空间与时间相对较少，然而自解释模型由于其对于模型决策的展示，及其对于模型内部透明化的能力，对于决策者是必不可少的. 由此，我们选择使用自解释的方法在图分类任务中进行模型设计，使模型在预测图分类的同时对于自身做出解释.

Lin 等人[13]的工作将基于图分类任务的 GNN 模型可解释的目标概括为提取与预测结果最相关的子图. 因此，旨在同时完成图分类任务和自解释任务的 GNN 模型的目标是在输出图分类结果的同时，输出与预测结果高度相关的子图. 上述解释方法的目标都是获取这一子图. 然而，现有解释方法未能充分捕获图结构的层次性信息，包括图结构中隐藏的等级、类别、重要拓扑结构及功能模块等信息. 这些信息对于图分析起到了关键作用. 图 1 展示了一个简易的员工网络，员工 B 与员工 C 处于同一工作组 A，员工 A 与员工 B 和员工 C 不处于相同工作组但处于同一工作部门 A，图中可以展示其层次信息，员工 B 与员工 C 由于其不直接相连，如果使用普通的解释模型，难以发现其层次信息. 而由于员工 A 与员工 B 和员工 C 由于拓扑距离较远，其处于同一公司部门的特征容易被模型忽略. 因此，进行层次化的分析，可以全面获取数据的隐藏信息，并提升 GNN 模型的可解释性.

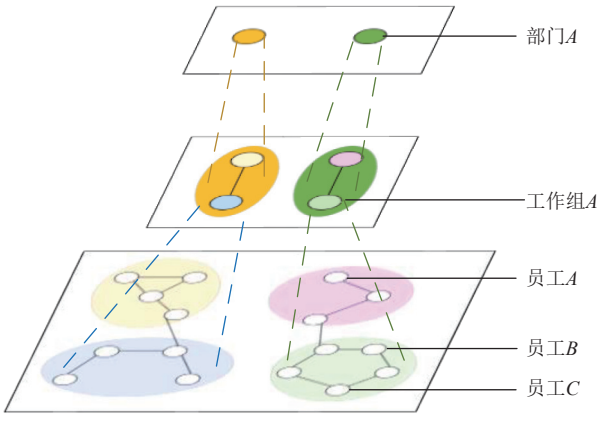


Fig. 1 An example of hierarchical structural information in graphs

图 1 图结构层次信息示例

图分类分层解释可以通过获取不同层次的解释子图(如用户感兴趣的商品的分类层次等)，综合考虑图中的局部和全局信息，将复杂的关联关系以层次化的形式展现出来. 此外，分层解释能够提供更好的数据可视化方式，方便决策者理解模型层次级的决策原因. 例如，在社交网络中对用户的社会地位进行分类和分层解释(如：用户职业、所在工作组、所在工作单位、所属地区、所在城市等)，可以帮助决策者确定不同用户所属的不同层次的社群，并据此做出决策调整. 由此，如何在输出图分类结果的同时，输出层次化的与预测结果相关的子图，成为实现图分类分层自解释的关键问题.

虽然，已有的 GNN 自解释模型(SEGNN, CAL, Orphicx 等)能够从不同视角提升图分类任务的可解释性，但对于基于层次的自解释图表示学习模型的设计仍然是缺乏的. 目前的图表示学习解释方法均不能捕获图数据的层次信息，针对该问题，本文提出了一种层次感知的自解释图表示学习模型，该模型在获取图信息并进行图分类的同时，层次化地对预测结果进行自解释. 基于以上，本文的工作具体贡献有 4 点：

1)针对 GNN 解释模型缺乏层次感知的问题，我们创新性地提出了一个可自解释的图表示学习模型 HSEGRL，通过引入层次解释的概念为图学习提供一种全新的解释视角；

2)基于单层解释信息的提取及分析，设计了解释子的基本结构，首先对图特征进行编码，并层次化地筛选解释子图，随后使用解码器对解释子图进行处理，来实现自解释子图的提取及其解释特征的挖掘，从而层次化地展示了图模型的决策原理；

3)针对提取层次解释的需求，对解释子中的池化层进行了优化，提出了一种自解释感知的池化操作. 该池化综合考虑了图拓扑与图节点特征所隐藏的信息，能够全面且深入地对模型进行解释.

4)在分子、蛋白质、社交数据等领域的图分类任务上进行了大量实验，实验结果表明了 HSEGRL 的有效性. 并与最新的 GNN 模型相比，其准确率存在明显提升. 此外，通过对解释结果进行可视化，可直观地说明分层解释的效果.

1 相关工作

在本节，我们将介绍一些与本文工作相关的技术，包括 GNN 可解释性技术、图池化方法. 我们分别介绍了这 2 种技术的主要发展路线，并归纳总结现有方法的优点及不足.

1.1 可解释图神经网络

在本节，我们将介绍可解释 GNN 的研究现状. Kakkad 等人[7]通过总结已有的 GNN 可解释性工作将可解释 GNN 分为事后解释 GNN 与自解释 GNN. 其中事后解释并没有将可解释性架构内置到模型中，

而是通过训练完成的模型的输入输出去预测其决策原因.

事后解释模型可以划分为: 基于分解的方法(decomposition-based method)、基于梯度的方法(gradient-based method)、基于代理的方法(surrogate-based method)以及基于扰动的方法(perturbation-based method). 基于分解的方法将模型的预测视为一个分数, 分解并逐层向回分配, 直到达到输入层. CAM[14] 旨在构建对具有全局平均池化(global average pooling, GAP)层和全连接层作为最终分类器的 GNN 的解释, 通过全局平均池化层的输出结果, 计算输入节点的重要性分数. 基于梯度的方法认为梯度代表变化的速率, 而预测关于输入的梯度表示预测对于输入的敏感程度. 基于代理的方法在局部范围使用简单的解释目标模型, 避免了在大范围选取输入值的情况下输入输出关系复杂的问题. GraphLime[15] 模型通过使用一种非线性代理模型 HSIC(Hilbert-Schmidt independence criterion) Lasso 对原模型进行局部解释. 基于扰动的方法利用扰动输入来寻找重要的子图作为解释, 较为经典的 GNNExplainer 与 PGExplainer 都是基于此方法.

由于事后解释模型主要依据已完成训练的模型的输入输出预测模型的决策过程, 具有很强的不确定性及不稳定性, 即模型不能很直观地代表原模型的决策过程, 而当输入输出受到微小的扰动时, 会使解释模型发生很大的变化. 因此, 能够反映模型内在的自解释模型受到越来越广泛的关注.

自解释模型方法主要通过捕获图中的隐藏信息, 如因果(causal)信息、结构信息、概念(concept)信息等来实现模型的自解释. CAL[11] 模型将因果机制引入 GNN 中, 实现了图模型的自解释, 即通过将图划分为因果相关图与无关图, 完成了对于图中因果信息的挖掘, 从而能够直观地理解模型决策原因. 然而, 由于 CAL 同样忽略了图上的层次信息, 会导致一些有效信息被划入无关图中, 而这些信息可能在一些层次上对于结果是有因果影响的. KerGNN[16] 将图核与 GNN 结合起来, 通过将可训练的隐藏图作为图滤波器, 从而发现图中的隐藏结构信息. 然而, 该方法只考虑到了图隐藏的结构信息, 而节点特征所隐藏的信息被忽略. Han 等人[17] 的工作创新性地将概念信息引入 GNN 解释工作中, 设计一种新的评价指标, 旨在通过多图学习捕捉图中的关键概念. 然而, 这种方法未能充分考虑概念信息的层次性, 例如员工、工作组、部门与公司之间的层次关系. 在低层次图中, 由于节点间距离较短, 高层次的重要概念(如员工所属的公司)可能会被忽略. 为了解决这一问题, 我们提出了一种自解释的 GNN 模型, 该模型能够感知并利用层次信息, 从而更准确地解释图分类决策.

在构建层次化的图解释模型时, 同样面临着一系列挑战. 首先, 图结构的复杂性增加了挖掘与分类结果相关的层次子图的难度, 处于同一层次的节点可能并没有直接相连, 导致这些层次信息难以被发掘和利用; 其次, 如何抽象不同角度的层次解释信息也为层次化的图解释模型带来了挑战, 对于不同层次的图拓扑与图特征信息的抽象方法决定了不同层次解释信息的提取结果, 因此, 设计多角度层次解释的有效抽象方法至关重要; 此外, 整合多尺度信息也是一个技术难题, 在获取了图拓扑与图特征信息后, 如何将这些处于不同维度的特征进行转换与分析也给层次解释带来挑战; 同时, 如何有效应用多层次解释信息也增加了模型构建的复杂性. 这些因素共同构成了设计高效层次化图表示学习模型的主要挑战. 因此, 探索创新性的方法来提升 GNN 在层次解释方面的表现是十分必要的.

1.2 图池化

近年来, GNN 已在图分类任务中取得较好的效果[18]. 随着 GNN 技术的快速发展, 越来越多的 GNN 工作引入池化方法来优化图分类任务. 图池化方法主要用于学习给定图的图粗化表示, 实现图信息的层次化处理, 将具有节点表示的输入图压缩为更小的图或整体图级表示, 以解决 GNN 捕获图子结构信息能力有限的问题.

GNN 的池化方法分为全局池化与局部池化. 全局池化算子通过将输入向量加和、取平均或取最大, 直接将整个输入转化为一个固定的向量; 局部池化又称层次池化, 通过提供图结构的多级表示, 使 GNN 逐级提取更高层次的图属性, 将一个图转化为多个表示向量. Top-k 池化操作[19-21] 提出了层次池化的概念, 通过使用节点特征和局部结构信息设计了一种池化方法对结构信息进行评分, 在每个池化步骤中只传播得分最高的前 k 个节点.

Grattarola 等人[22] 的工作详细归纳总结了 GNN 中池化操作的方法, 提出了一个图池化操作的通用框架, 并将其划分为 3 个部分: 选择(selection)、归约(reduction)和连接(connection). 选择操作的作用是筛选节点, 可以归纳为基于聚类的和基于节点删除的方法. 其中, 基于聚类的方法通过将有相似特征的节点聚合为一个新节点, 实现对图结构层次信息的利

用，能够更好地发掘图中节点的相似性信息；而基于节点删除的方法仅保留对于结果最重要的节点，从而提取更重要的信息，这对于处理大规模图更有优势. 当我们得到原始粗化版本的图后，该图仅具有筛选过的节点，而规约操作的目的就是获得这些筛选过的节点的新表征，随后的连接操作就是将这些新节点重新连接为图.

典型的池化工作为：DiffPool[23]设计了一个差分池化算子，该算子通过在每个图卷积层学习软聚类，将节点分配到某一类中. 由于DiffPool会引入较多参数，后续很多工作会对其进行优化. SAGPool[24]利用基于GNN的自注意力机制来学习节点得分，并通过对节点得分进行排序来选择节点. 尽管这些GNN的图池操作在图分类任务上已经取得了较好的结果，它们通常利用图学习的方法来筛选子图，这使得中间层次决策过程对于结果是不可见的，因此缺乏可解释性. 由此，CommPool[25]提出了一种通用的可解释的池化方法，通过在每个层次内运行PAM（partitioning around medoids）算法，对节点特征相似性进行评分，从而使结果具有可解释性. 然而该方法仅关注了特征相似性，而忽略了图结构特征，同时该方法仅对提取的子图进行了池化，忽略了提取子图的隐藏信息.

由于图池化方法能够实现对于图数据的分层处理，我们的模型通使用该方法实现模型的层次感知. 此外，由于现有的池化不能同时进行图结构及特征的自解释，我们的模型针对这一问题提出了解释感知池化（explanation-aware pooling, EAPooling）方法.

2 方　　法

在本节，我们首先进行问题的描述，定义相关的数学表达式及变量. 随后将介绍提出的HSEGRL模型，该模型主要由解释子构成，而解释子主要由3部分组成：GNN编码器、解释感知池化层和解释子图解码器. 我们首先详细介绍层次自解释模型的架构，随后说明了解释子的具体结构，最后描述解释感知池化层的相关算法.

2.1 问题描述

首先，我们将明确本文GNN的数学表达. 用$G=\{\boldsymbol{A},\boldsymbol{X}\}$来表示一个图，其中$\boldsymbol{A}\in\mathbb{R}^{N\times N}$是图G的邻接矩阵，$A_{i,j}=1$表示节点$i$和节点$j$之间存在一条边. G具有N个节点，$\boldsymbol{X}\in\mathbb{R}^{N\times d}$表示节点特征矩阵，图$G$中的每个节点都可以用一个$d$维向量来描述.

Gilmer等人[26]使用消息传递框架总结了GNN. 他们的工作介绍了一种将GNN模型集成到MPNN框架中的方法，从而实现了各种GNN模型的实现. MPNN包含2个阶段：聚合阶段和组合阶段，对应操作如式（1）所示. 在聚合阶段，图中的每个节点从其相邻节点中收集信息，目的是捕捉局部邻域中的相关信息. 在组合阶段，MPNN将所有邻居的聚合表示与自身表示相结合，生成一个准确表示该节点的新表示.

$$\begin{cases}\boldsymbol{a}_n^{(k)}=AGGREGATE^{(k)}\left(\boldsymbol{z}_u^{(k-1)}:u\in N(n)\right),\\ \boldsymbol{z}_v^{(k)}=COMBINE^{(k)}\left(\boldsymbol{z}_n^{(k-1)},\boldsymbol{a}_n^{k}\right),\end{cases}\tag{1}$$

其中，$\boldsymbol{z}_v^{(k)}$表示第k层节点v_n的节点特征，$N(n)$代表图G的邻居节点n，$AGGREGATE^{(k)}(\cdot)$及$COMBINE^{(k)}(\cdot)$分别是聚合阶段和组合阶段消息传递机制的组成部分. 由于MPNN方法可扩展性强且性能高，多数GNN模型的实现都基于这一框架，因此我们的模型也基于这一框架.

Lin等人[13]的工作将图分类解释的目标总结为获取一个与预测结果最相关的子图. 基于这一总结，我们将优化目标定义为：给定一个图分类任务，在完成图分类任务的同时，输出与预测结果相关的层次化子图. 其目标是需要准确而全面地提取图层次解释信息的同时，将解释信息用于优化图分类任务. 接下来我们将详细介绍该方法的具体实现.

2.2 模型框架

为了实现分层解释的目标，即在进行图分类的同时，层次化地输出与预测结果最相关的子图. 本文创新性地提出了解释子的概念，作为获取图结构单层解释的基本模块. 使用每个解释子捕获一层解释信息，通过多层解释子的串联，挖掘多层与分类结果相关的子图. 本文首先介绍HSEGRL模型的整体框架，具体结构如图2所示，其中解释子及池化的具体操作将在后续章节中介绍，本节将主要介绍HSEGRL模型如何综合考虑多层解释信息，并将其用于优化图分类结果.

我们以3层解释为例来介绍该框架的运行过程. 首先，将3个解释子串联，并将EAPooling层的输出内容，即解释相关子图特征，作为下一层的输入，通过3层解释子串联获取图分类结果，图分类任务的损失函数$\mathcal{L}_{\mathrm{GC}}$使用交叉熵损失函数作为最终的损失，其公式为：

$$\mathcal{L}_{\mathrm{GC}}=-\sum_i 1_{\{y=i\}}\log p_i,\tag{2}$$

其中，$1_{\{y=i\}}$为指示函数，当$y=i$时输出为1，反之，输出为0，p_i为$p_i=\mathrm{e}^{z_i}/\sum_j \mathrm{e}^{z_j}$.

与此同时，还需对获取的层次化解释信息进行

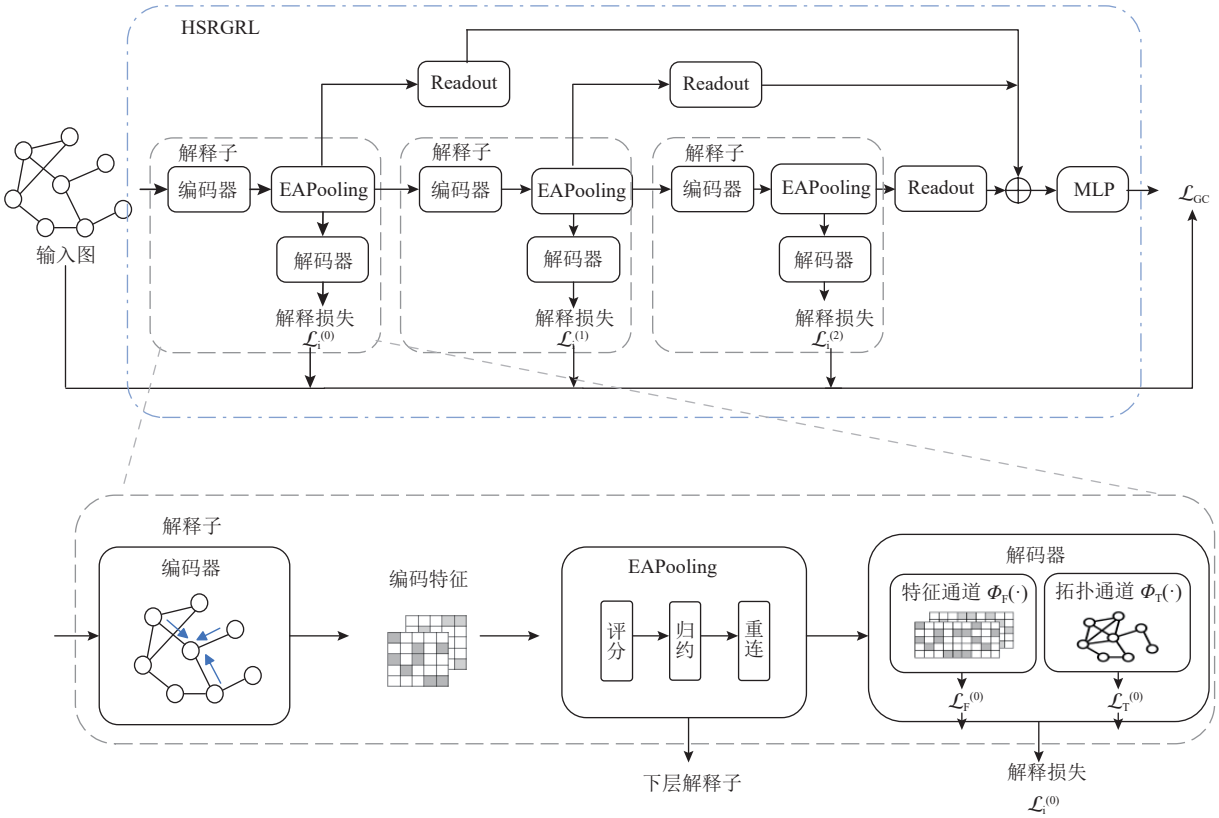


Fig. 2 Framework diagram of HSEGRL

图 2 HSEGRL 框架图

处理，我们将每层的 EAPooling 输出结果同时输入到解码器及下一层解释子中，通过解码器的输出计算该层的解释损失，经过 3 层解释子后，将解释损失相加，实现对于解释信息的调整.

解码器的目的是获取解释图的高阶信息，为了综合考虑原图中隐藏的结构与特征解释信息，我们重新设计解码器，将其分为特征通道 $\phi_F(\cdot)$ 与拓扑通道 $\phi_T(\cdot)$，使之能够同时感知隐藏的结构与特征信息，其具体实现在 2.3 节中介绍.

在拓扑损失函数中，使用 L_2 正则化使得解码后的解释图的拓扑结构尽可能多地保留原图信息，以实现对于解释相关子图的结构及特征调整，对应每一层解释相关模型的拓扑损失函数为：

$$\mathcal{L}_{\mathrm{T,interp}}^{(l)} = \left\| \boldsymbol{A}_{idx^{(l)}} - \phi_{\mathrm{T}}\left(\boldsymbol{Z}^{(l)}\right) \right\|_{\mathrm{F}}^{2}, \tag{3}$$

其中，$\boldsymbol{A}_{idx^{(l)}}$ 表示第 l 层经筛选后的邻接矩阵，l 表示解释的层次，$idx^{(l)}$ 表示第 l 层选取的子图节点索引，获取方法将在池化操作中详细介绍，$\phi_T(\cdot)$ 表示拓扑通道解码器，$\mathcal{L}_{\mathrm{T,interp}}^{(l)}$ 为拓扑通道损失函数，其中下标 T 表示拓扑通道，而下标 interp 表示解释感知.

而特征损失函数 $\mathcal{L}_{\mathrm{F,interp}}^{(l)}$ 为：

$$\mathcal{L}_{\mathrm{F,interp}}^{(l)} = \left\| \boldsymbol{Z}_{idx^{(l)}} - \phi_{\mathrm{F}}(\boldsymbol{Z}^{(l)}) \right\|_{\mathrm{F}}^{2}, \tag{4}$$

其中，下标 F 表示特征通道，interp 表示解释感知，$\phi_F(\cdot)$ 表示特征感知的解码器，$\boldsymbol{Z}^{(l)}$ 表示点的特征，$\boldsymbol{Z}_{idx^{(l)}}$ 表示第 l 层提取的解释感知子图原特征.

由此，我们获取了每层解释子图的损失函数，接下来我们 2 个通道的损失函数结合起来得到第 l 层的解释感知损失函数 $\mathcal{L}_{\mathrm{interp}}^{(l)}$：

$$\mathcal{L}_{\mathrm{interp}}^{(l)} = \alpha_{\mathrm{F}}^{(l)} \mathcal{L}_{\mathrm{F,interp}}^{(l)} + \alpha_{\mathrm{T}}^{(l)} \mathcal{L}_{\mathrm{T,interp}}^{(l)}, \tag{5}$$

其中，α 是拓扑与特征通道自适应参数，其具体实现将在 2.3 节中详细介绍.

在获取图分类损失函数与可解释损失函数后，可以获得完整的模型损失函数，模型损失函数 $\mathcal{L}_{\mathrm{total}}$ 由式(6)表示：

$$\mathcal{L}_{\mathrm{total}} = (1-\lambda_1-\lambda_2-\lambda_3)\mathcal{L}_{\mathrm{GC}} + \lambda_1 \mathcal{L}_{\mathrm{interp}}^{(0)} + \lambda_2 \mathcal{L}_{\mathrm{interp}}^{(1)} + \lambda_3 \mathcal{L}_{\mathrm{interp}}^{(2)}. \tag{6}$$

在完成所有的损失函数运算后，需要将每层解释子中 EAPooling 的分层结果(即每层学习到的节点特征)分别输入到一个读出(Readout)层进行相加，经过 1 个 MLP 层后，得出图的分类预测结果. Readout 层的作用是提取解释子图的特征，并将其与节点特征拼接为固定大小，便于后续计算，表达式为：

$$s = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{z}_i \left\| \max_{i=1}^{N} \boldsymbol{z}_i, \right. \tag{7}$$

其中，N表示子图中节点的数目，z_i表示节点i的特征，$\|$表示张量拼接操作.

基于以上方法，我们实现了使用解释子构成分层自解释图学习模型. 通过设计对应的损失函数，实现了图结构与图节点特征相关解释子图的分层提取及调整，同时将解释内容用于指导图分类任务的结果.

2.3 解释子设计

我们已经完成了对模型整体框架的介绍，以及如何利用解释子构成层次化自解释图表示学习模型，在本节我们将主要介绍解释子的具体功能和框架.

解释子由编码器、EAPooling、解码器构成. 每一层解释子通过编码器获取图节点的表征，然后利用EAPooling提取结构与特征相似的节点，以提取该层结构与特征上的重要子图. 随后，通过解码器以获取该层解释子图并提取其高阶信息. 解释子通过对编码器的训练，实现了图表示学习的分类任务，并通过EAPooling进一步提取结构与特征层面上的重要子图，这些子图反映了模型学习到的决策模式. 解码器通过训练表征了模型学习到的信息，使得模型在执行图分类任务的同时能够进行自我解释. 通过多层解释子的堆叠，可以在上层解释子提取的信息基础上进一步提取层次信息，每一层都代表了数据的不同抽象层次. 综合考虑不同层次的表征结果与解释信息，使得HSEGRL模型能够对数据进行层次化处理，同时层次化地学习自身的表达方式. 整个表示过程是直观的且可解释的，由此实现了HSEGRL模型的层次自解释能力.

解释子作为实现层次图表示学习解释模型的单元，其目的是从拓扑角度与特征角度捕获一层解释子图并表征其隐藏信息，整体算法如算法1所示.

算法1. 解释子算法.

输入： 图$G=\{\boldsymbol{A},\boldsymbol{X}\}$，解释子层数$n$，评分函数$S_l$，特征编码函数$f(\cdot)$，解码器$\phi(\cdot)$，损失函数$f_{\text{loss}}(\cdot)$；

输出： 解释子参数.

① for $l=1$ to n: do

② 特征编码：$\boldsymbol{Z}\leftarrow f(\cdot)$；

③ 节点重要性评估：$Score^{(l+1)}\leftarrow S^{(l)}(\boldsymbol{A},\boldsymbol{Z})$；

④ 节点筛选：$idx^{(l)}=TOP_k(\cdot)$；

⑤ 子图与特征重构：

$$\boldsymbol{Z}^{(l+1)}=\boldsymbol{Z}^{(l)}_{idx^{(l)}}\odot Score^{(l)}_{idx^{(l)}},$$
$$\boldsymbol{A}^{(l+1)}=\boldsymbol{A}^{(l)}_{(idx^{(l)},idx^{(l)})};$$

⑥ 解码：

$$\boldsymbol{Z}^{(l+1)}_{interp}=\phi_{\text{F}}(\boldsymbol{A}^{(l+1)},\boldsymbol{Z}^{(l+1)}),$$
$$\boldsymbol{A}^{(l+1)}_{interp}=\phi_{\text{T}}(\boldsymbol{A}^{(l+1)},\boldsymbol{Z}^{(l+1)});$$

⑦ 更新损失函数：$\mathcal{L}\leftarrow f_{\text{loss}}(\cdot)$；

⑧ end for

首先我们进行特征编码(行①)，给定一个基于GNN的编码器$f(\cdot)$及图$G=\{\boldsymbol{A},\boldsymbol{X}\}$对节点特征进行编码，其中GNN编码器可以是GCN[27]，GAT[20]，GIN[28]等图分类模型，由此可以得到节点的特征表示：

$$\boldsymbol{Z}=f(\boldsymbol{A},\boldsymbol{X}). \tag{8}$$

随后，将获得的节点表示与图数据输入到EAPooling层后(行③~⑤)，根据获取的节点编码与图拓扑信息分别对节点进行特征及拓扑重要性评估，并筛选出最重要的节点. 然后更新该层次的子图及其特征表示，并将其输入到下一层解释子. 与此同时，将该结果输入到该解释层的解码器(行⑥)，以便调整解释结果，并充分利用解释结果来优化分类任务的决策. 为了使解码器能够充分考虑结构信息与特征信息，我们将解码器分为特征感知通道与拓扑感知通道. 其中，特征感知解码器的作用是获取层次解释子图的隐藏特征，具体实现如式(9)所示. 我们使用简单的GCN作为解码器，其中$\boldsymbol{Z}^{(l)}$表示第l层的解释感知特征：

$$\phi_{\text{F}}\left(\boldsymbol{Z}^{(l)}\right)=GCN\left(\boldsymbol{Z}^{(l)},\boldsymbol{A}\right). \tag{9}$$

而拓扑感知解码器的目的是挖掘层次解释子图中所隐藏的高阶拓扑结构信息，具体实现为：

$$\phi_{\text{T}}\left(\boldsymbol{Z}^{(l)}\right)=Relu(\boldsymbol{Z}^{(l)^{\text{T}}}\boldsymbol{Z}^{(l)}), \tag{10}$$

其中，Relu函数用于确保结果保持非负.

该特征感知与拓扑感知的解码器能够评估解释子图中节点间的解释特征相似度，更好地利用所获取的解释子图与解释特征. 随后，将详细描述EAPooling的结构，其通过筛选解释相关子图，提取图结构感知与特征感知的层次解释信息表征.

2.4 EAPooling方法

为了综合考虑节点特征与拓扑信息，我们提出了一种可以自解释的图池化方法——EAPooling方法，使获取的层次解释子图能同时考虑到图节点特征与图拓扑所隐含的信息. 这2类信息都被先前的工作证实是有价值的，上文提到的CommPool[25]仅考虑了节点特征相似性，而HoscPool[29]模型仅考虑了图拓扑信息，这2类信息都能对图分类的准确率进行一定的优化. 因此，为了更全面地解释图分类任务，我们设计了一种分别获取图节点特征信息解释重要性与图拓扑信息解释重要性的方法，通过自适应地调整二者权重，将二者结合起来综合评估其解释重要性.

本文的方法基于Grattarola等人[22]工作总结的选

择、规约、连接的框架，首先在选择阶段，综合考虑图节点特征及图拓扑特征. 对于节点特征信息的获取，EAPooling 使用 MLP 方法识别每个节点特征的重要性，对于分层解释的第l层，我们可以获得特征解释重要性评分：

$$Score_{\mathrm{F}}^{(l)}=\delta\left(MLP\left(\boldsymbol{Z}^{(l)}\right)\right), \tag{11}$$

其中，$MLP(\cdot)$作为全连接层评估节点特征解释重要性，而$\delta(\cdot)$代表激活函数，通过这个方式能过获取节点特征重要性.

对于图拓扑重要性，EAPooling 使用结构注意力机制来评估，结构注意力系数如式(12)所示.

$$\alpha_{i,j}=Softmax_i\left(sim_{i,j}\right)=\frac{\exp\left(sim_{i,j}\right)}{\sum\limits_{v_k\in N(v_i)}\exp\left(sim_{i,j}\right)}, \tag{12}$$

其中，$Softmax_i(\cdot)$函数为节点归一化函数，$N(v_i)$表示节点v_i的邻居节点. $sim_{i,j}$表示节点i与节点j之间的拓扑相似性关系，其数学表示如式(13)所示，其中$|\cdot|$表示节点个数. 通过节点邻居相似度表征拓扑相似度，邻居重合度越高，其拓扑相似度越高，通过这种方式评估节点的重要性.

$$sim_{i,j}=\frac{\left|N(v_i)\cap N(v_j)\right|}{\left|N(v_i)\cup N(v_j)\right|}. \tag{13}$$

在获取了图结构注意力系数$\alpha_{i,j}$后，将其应用于第l层节点拓扑重要性评分$Score_{\mathrm{T}}^{(l)}$：

$$Score_{\mathrm{T}}^{(l)}=\delta\left(\sum_{N_i}\alpha_{i,j}\boldsymbol{W}\boldsymbol{z}_j\right), \tag{14}$$

其中，$\boldsymbol{W}$为训练权重并与式(16)共享权重，$\boldsymbol{z}_j$为节点j的特征向量表示.

综上，评分机制使用 MLP 方法评估每个节点的特征解释重要性，并通过引入结构注意力机制评估每个节点的拓扑重要性，得到每个节点的解释重要性评分$Score_{\mathrm{T}}^{(l)}$与$Score_{\mathrm{F}}^{(l)}$，之后使用自适应的方法将这 2 个评分进行组合，即对这 2 个分数赋予自适应的权重，分别为$Att_{\mathrm{F}}^{(l)}$，$Att_{\mathrm{T}}^{(l)}$，两者之和为 1. 我们统一用$Att_{\mathrm{X}}^{(l)}$表示. 自适应权重通过对图拓扑信息注意力系数的获取，确保拓扑相似度大的节点间能够聚合更多的结构信息.

$$Att_{\mathrm{T}}^{(l)}=Softmax_i\left(e_{i,j}\right)=\frac{\exp\left(e_{i,j}\right)}{\sum\limits_{v_k\in N(v_i)}\exp\left(e_{i,j}\right)}, \tag{15}$$

其中$e_{i,j}$表示注意力，其表达式为：

$$e_{i,j}=\delta\left(a\left(\left[\boldsymbol{W}\boldsymbol{z}_i\,\|\,\boldsymbol{W}\boldsymbol{z}_j\right]\right)\right), \tag{16}$$

其中，$a(\cdot)$表示单层神经网络，用于将向量映射为一个数以表征节点间的重要性，$\|$表示张量的拼接，$\boldsymbol{W}$是可训练的参数.

由此，我们能获取$Score^{(l)}$评分最终表示为：

$$Score^{(l)}=Att_{\mathrm{F}}^{(l)}\odot Score_{\mathrm{F}}^{(l)}+Att_{\mathrm{T}}^{(l)}\odot Score_{\mathrm{T}}^{(l)}, \tag{17}$$

其中l表示解释层数，$\odot$表示广播逐元素乘积(broadcast element-wise product).

随后在归约阶段，EAPooling 需要选取最重要的k个节点，通过综合考虑图拓扑信息与结构信息获取当前层次与图分类最相关的子节点，其 top-k 评分选取机制表示：

$$idx^{(l)}=TOP_k\left(Score^{(l)}\right), \tag{18}$$

其中，$TOP_k(\cdot)$通过将$Score^{(l)}$值进行排序，并返回其中最大的k个值的索引. 而$idx^{(l)}$表示在第l层保留的k个解释子图节点的索引.

在获取筛选后的节点后，需要将这些节点重新连接为图，式(19)实现连接图这一操作，并更新该层次的节点特征：

$$\begin{cases}\boldsymbol{Z}^{(l+1)}=\boldsymbol{Z}_{idx^{(l)}}^{(l)}\odot Score_{idx^{(l)}}^{(l)},\\ \boldsymbol{A}^{(l+1)}=\boldsymbol{A}_{(idx^{(l)},idx^{(l)})}^{(l)},\end{cases} \tag{19}$$

其中$\boldsymbol{Z}_{idx^{(l)}}^{(l)}$表示经过筛选后的第$l$层节点特征向量矩阵，$\boldsymbol{A}_{(idx^{(l)},idx^{(l)})}^{(l)}$表示在原矩阵上根据筛选后节点的索引过滤后得到的邻接矩阵. $\boldsymbol{Z}^{(l+1)}$与$\boldsymbol{A}^{(l+1)}$分别表示传输到$l+1$的层的节点编码特征及邻接矩阵.

通过上述方法，我们实现了单层的解释节点提取，并将其继续输入到下一层进一步评估节点特征重要性与结构重要性来选取该层次最具有解释性的子图.

综上，我们完成了 HSEGRL 具体实现方案的介绍，该模型通过设计的解释子结构，能够在每 11 层捕获结构和特征的层次化解释. 通过多层解释子的堆叠，HSEGRL 实现了对图数据的多层次解释. 在解释子内部包含了编码器、解释感知池化及解码器，这些组件共同工作，实现模型预测的同时进行模型自解释，并将捕获的图数据层次信息用于优化图分类任务的决策.

3 实验与评估

本节将阐述对 HSEGRL 模型进行的相关实验及评估. 首先，在常用的分子、蛋白质与社交网络数据集上进行图分类准确率实验，并与相关领域的最新工作进行对比. 随后，将实验结果进行可视化分析，以证明分层解释的有效性. 接着，进行参数实验，说

明模型参数对于实验结果的影响. 最后, 对模型进行时间复杂度分析, 并验证其在较大规模数据集上的实验结果.

3.1 数据集与基准模型

在数据集方面, 我们分别在分子数据集与社交网络数据集进行实验验证, 对于分子数据集, 我们选择了 MUTAG[30], NCI1[31]. 其中, MUTAG 是一组硝基芳香化合物, 实验目标是预测这些化合物是否具有诱变效应, NCI1 与抗癌药物筛选有关, 化学物质被评估为对肺癌细胞呈阳性或阴性. 对于蛋白质数据集, 我们选择了 DD[30-32] 与 PROTEINS[33]. 其中 DD 在蛋白质数据库的非冗余子集中抽取了了 1 178 个高分辨率蛋白质, 其中节点是氨基酸, 如果 2 个节点之间的距离少于 6 埃(Angstroms), 则用 1 条边连接; PROTEINS 是一个蛋白质数据集, 其中的蛋白质被分类为酶或非酶. 节点代表氨基酸, 如果节点之间的距离小于 6 埃, 那么这 2 个节点之间会有 1 条边连接. 而对于社交数据集, 我们选择了 COLLAB, IMDB-BINARY, IMDB-MULTI[34], 其中, COLLAB 是 1 个科学合作数据集, 每个图对应于 1 个研究人员的自我网络, 即研究人员及其合作者是节点, 边表示 2 位研究人员之间的合作关系. 1 个研究人员的自我网络有 3 种可能的标签, 即高能物理、凝聚态物理和天体物理; IMDB-BINARY 与 IMDB-MULTI 是电影合作数据集, 其具体内容将在 3.3 节中进行介绍.

此外, 我们还在大数据集上进行实验测试, 数据集具体信息如表 1 所示.

在选择模型时, 我们分别选择了典型的池化模型和图分类模型与可解释模型进行对比.

1)对于池化模型, 我们选择了 DiffPool[23], SAGPool[24], Top-k[19-21], SortPool[35], HoscPool[29], 这些方法通过捕获隐藏的图结构, 并使用池化方法提取这些结构的特征表示以提高图分类的准确率;

2)在 GNN 领域, 我们选择了 GIN[28], GSN[36], GNN-AK+[37] 模型, 这些模型通过捕获高阶图结构信息和优化模型的表达能力来提高图分类的准确率;

3)对于可解释 GNN 模型, 我们选择了 kerGNNs[16], ProtGNN[38], CAL[11], GIB[39-40], 这些模型通过将图核、原型、因果、信息熵等概念引入解释模型, 使这些模型可以从不同角度实现图分类的同时进行模型自解释.

Table 1 Graph Classification Datasets

表 1 图分类数据集

数据集	图数	类别数	平均节点数	平均边数
MUTAG	188	2	17.93	19.79
PROTEINS	1113	2	39.06	72.82
NCI1	4110	2	29.87	32.30
DD	1178	2	284.32	715.66
COLLAB	5000	3	74.49	2457.78
IMDB-BINARY	1000	2	19.77	96.53
IMDB-MULTI	1500	3	13.00	65.94
MCF-7	27770	2	26.40	28.53
SW-620	40532	2	26.06	28.09
YeastH	79601	2	39.45	40.75

HSEGRL 模型的超参数设置为: 损失函数权重设置为 $\lambda_1 = 0.2, \lambda_2 = 0.2, \lambda_3 = 0.2$, 池化率设为 0.75, 编码模型选择 GCN 模型. 为了与 HSEGRL 实验设置一致, 对于所有的池化对比模型, 均使用了 3 层池化, 池化率设为 0.75. 为保持 GNN 模型一致性, 所应用的 GNN 基线模型均基于消息传递 GNN 框架, 实验中也使用了 3 层 GNN. 对于可解释 GNN 模型, 同样采用了 3 层模型进行模型解释. 批大小 batch_size 设为 256, 训练轮数 epoch 设为 200, 学习率设为 0.0005. 其余模型参数保持原文中最佳设置. 所有实验重复执行 5 次以计算均值和误差. 实验硬件平台为英伟达 A800 GPU, 软件平台为: CUDA 12.2, Python 3.9, Pytorch-geometric 2.4.0.

3.2 图分类实验结果分析

图分类准确率结果如表 2 所示, 我们的结果在分子图、蛋白质图与社交图中都取得了较好的结果, HSEGRL 对比 12 个基准模型都有较为明显的优势. 其中, 在 PROTEINS, NCI1, DD 数据集上, HSEGRL 模型均优于基准模型. 特别是在较大的数据集 DD 上, HSEGRL 相对于最好的基准模型平均相对增益为 2.10 个百分点. 然而, 在最小的数据集 MUTAG 上, HSEGRL 未能取得最好的结果. 因为 GNN-AK+与 GIN+CAL 模型能够更好地适应小数据集, 但 GNN-AK+及 GIN+CAL 模型由于计算量较大而不能在较大的数据集(如 DD)上完成分类任务. 尽管如此, HSEGRL 在 MUTAG 上相对于其它模型仍具有显著的优势. 此外, HSEGRL 在社交数据集上也取得了较好的结果. 尽管在 IMDB-MULTI 上效果比 GSN 略差, 但在 COLLAB 与 IMDB-BINARY 这 2 个数据集上取得了较好的结果, 在 IMDB-BINARY 上能够比最好的基准模型的分类准确率提高 2.2 个百分点.

上述结果证明了 HSEGRL 的有效性. 就基准模型而言, 图池化方法计算速度较快, 但效果略差于挖掘了图中高阶信息的 GNN 模型, 及挖掘了图中解释信息的图自解释模型. 而 HSEGRL 将这些信息集成

Table 2 Experimental Results of Graph Classification Accuracy

表 2 图分类准确率实验结果

方法	分子数据集		化学数据集		社交数据集		
	MUTAG	NCI1	PROTEINS	DD	COLLAB	IMDB-B	IMDB-M
DiffPool	83.31±5.28	77.78±2.25	73.81±3.35	75.45±3.56	67.73±2.37	68.21±2.91	45.78±3.73
SAGPool	75.85±3.41	67.45±1.11	73.35±2.32	76.45±0.97	78.43±6.67	74.35±4.41	47.54±5.21
Top-k	69.93±1.13	71.85±2.82	69.81±3.99	73.71±1.78	73.34±4.47	73.17±4.84	48.80±3.19
SortPool	86.17±7.53	75.48±1.62	77.00±1.68	77.47±1.87	77.84±1.22	73.00±3.50	49.53±2.29
HoscPool	87.13±5.57	73.84±3.38	74.44±4.48	78.66±2.24	78.85±5.58	75.57±4.43	51.58±3.32
GIN	78.65±5.33	68.30±1.04	67.85±3.57	66.87±2.31	79.35±5.61	71.21±6.67	48.47±3.20
GSN	90.16±6.57	83.25±4.33	75.62±6.65	78.87±5.43	81.45±1.41	75.84±3.42	**53.32±2.83**
GNN-AK+	**90.33±7.21**	83.12±1.96	76.47±4.74	-	-	76.54±2.34	52.21±3.35
GIB	78.85±6.32	62.27±7.82	76.96±6.60	73.35±8.87	74.25±1.83	72.28± 4.34	50.35±3.64
ProtGNN	80.50±9.12	75.53±4.42	72.32±3.54	69.85±3.33	-	-	-
GIN+CAL	88.87±8.64	78.84±3.12	76.75±2.35	-	82.26±1.25	74.17±3.21	52.60±2.36
KerGNN-3	86.75±4.63	80.25±1.26	75.35±4.85	75.24±4.96	73.12±2.36	75.15±5.57	52.13±4.75
HSEGRL	88.42±5.31	**83.53±2.03**	**77.15±2.27**	**79.97±3.11**	**82.85±2.08**	**77.85±1.15**	52.86±2.80

注：IMDB-B 代表 IMDB-BINNARY 数据集，IMDB-M 代表 IMDB-MULTI 数据集；“-”表示训练时内存溢出；“±”前后数字分别代表平均准确率和准确率误差；黑体数值表示结果最优.

起来，通过层次化的分析图特征及图拓扑信息，其分类准确率明显优于基准模型.

3.3 可视化验证

在本节我们进行了解释结果的验证，在 GNN 事后可解释模型中存在很多可量化的指标以证明解释[41]，但自解释模型缺乏应用广泛的量化指标，由此，我们基于 KerGNNs[16]，ProtGNN[38] 模型的可视化方法设计了 HSEGRL 可视化验证实验. 我们分别选择了分子数据集 MUTAG 与社交数据集 IMDB-BINARY 进行可视化实验，并可视化地输出了它们每层的解释结果. 其中，MUTAG 包含了芳香族和杂芳香族硝基化合物的化学结构信息，其分类表明了该化合物是否具有致突变性. 而根据文献 [30] 中的说明，层次堆叠的苯环与硝基基团都会导致化合物具有致突变性. 而 IMDB-BINAR 是一个电影协作数据集，其收集了 IMDB 上不同电影的演员/演员和类型信息. 对于每个图，节点表示演员/女演员，如果演员出演过同一部电影中，那么节点间存在一条边. 该数据集生成了动作和爱情题材电影的合作图，并为每个演员导出了自我网络. 图分类的任务就是识别一个自我网络图属于哪个流派. 该数据集的节点不包含节点 ID 及节点特征，仅通过不同的演员结构信息区别自我网络的类型.

实验中池化率参数设为 0.5，可视化结果如图 3 所示，在分子图的原始图中，不同颜色的节点代表不同的分子，而社交网络由于没有节点特征，因此全用

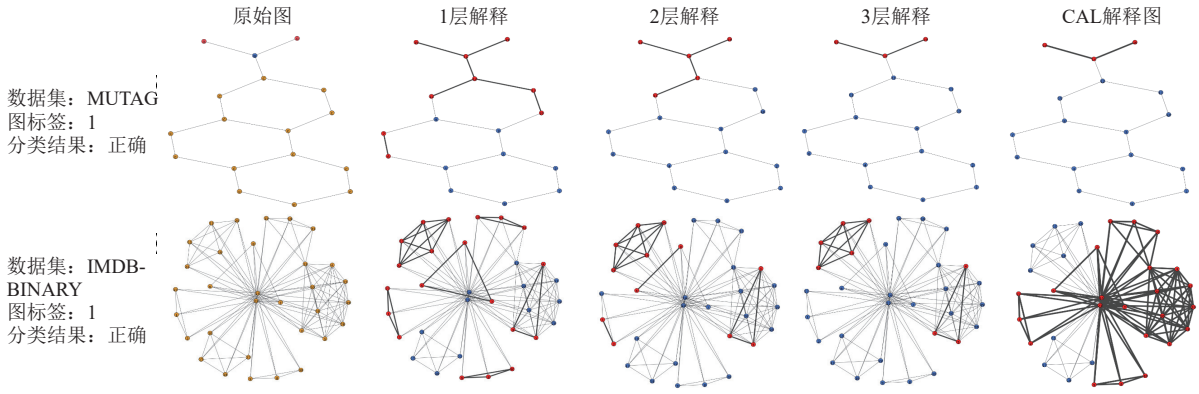


Fig. 3 Visualization of hierarchical explanation results

图 3 层次解释结果可视化

黄色表示. 在可视化的层次解释结果中, 加粗的边代表提取的解释子图边, 红色节点代表抽取的层次解释子图节点, 蓝色节点代表在该层与结果无关的节点. 图 3 最右侧的解释结果为比较模型 CAL 的解释结果.

而 HSEGRL 在低层次, 捕捉到了与硝基基团相连接的苯环部分结构, 随着解释层次的升高, 该模型能够更加精确地捕捉到硝基功能基团. 低层次能够使解释更具有泛化能力, 避免了模型由于只提取到了硝基基团, 但硝基基团并未与苯环相连, 从而产生误分类的情况. 而高层次由于其具有较准确的基团捕获能力, 能够提升模型分类的准确率.

在社交网络中, 我们可以看到 HSEGRL 在低层次不仅能够识别出稠密子图所对应的关键子图结构, 还能进一步发掘距离较远的关键子图. 这些子图在拓扑上并不相连, 这相较于一些只能提取强连通解释子图的模型具有很大的优势, 并更加符合决策者对于社交网络的理解. 而在高层解释中, 可以准确识别出 IMDB-BINARY 数据集中稠密子图的核心子图结构, 这一定程度上提高了该数据分类的精度.

在对比模型中, CAL 模型将因果机制引入 GNN, 该模型通过注意力模型, 将图数据中的因果图与无关图进行分离, 从而降低无关因素对分类任务的影响, 实现提取数据集中因果内容用于分类任务的目的. 该模型通过控制无关图参与输出的部分接近均匀分布, 以降低无关图在输出中的影响. 该模型能够有效提取图中区别于广泛存在信息的关键信息, 然而如图 3 所示, MUTAG 数据集中由于存在均匀分布的信息即苯环, 使得该模型能够有效提取硝基基团, 然而该方法仍然忽略了堆叠的苯环导致突变的特性. 相比之下, 在 IMDB-BINARY 数据集上, 由于缺乏均匀分布的无关信息, CAL 无法有效地提取关键信息. CAL 通过控制无关图参与输出的部分接近均匀分布, 导致模型提取无关图时, 容易提取数据集中均匀分布的部分, 即数据集中广泛存在的信息, 由于 CAL 降低无关图对于输出影响的特性, 忽略数据集中普遍存在的信息, 这些信息在社交网络等应用中通常更为重要. 同样地, GIB 模型通过分析图的信息瓶颈, 设计模型的损失函数, 确保学得的图表示形式仅含有必要的信息, 但也存在无法学得数据集中广泛存在的信息的问题, 导致在社交网络中应用效果不佳.

而 KerGNN 模型将图核方法引入 GNN 模型, 通过从数据集中提取图核信息, 并在分类任务中使用提取到的图核对目标图数据进行过滤, 实现解释模型学习到的决策模式的目的, 这使得该模型更加容易捕获数据集中广泛存在的信息, 从而忽略了关键信息. 类似地, ProtGNN 模型通过从数据集中提取原型信息后, 比较图数据与原型信息的相似度, 找到对该图最具有解释能力的原型, 该模型同样更加容易提取图信息中普遍存在的信息, 如 MUTAG 数据集中广泛存在的苯环, 从而忽略某些关键子图中的信息. 而 HSEGRL 模型采用分层解释的方法, 逐层提取图特征, 既能通过提取高层次子图信息提升模型的解释能力, 又能通过提取低层次图信息避免数据集中普遍存在的信息提取不足的问题, 同时避免了不同层次信息的混淆, 从而增强了模型的泛化能力.

综上, 基于层次解释子的 HSEGRL 模型在模型解释和模型精度上存在明显优势, 它能够在低层次提取图数据中的关键信息并增强解释的泛化能力, 并在高层次去除干扰的节点、边及子结构信息, 从而实现利用层次化的图拓扑及特征信息进行图分类的同时, 为决策者提供模型的层次化解释.

3.4 模型参数分析

在本节我们将评估相关参数对实验结果的影响, HSEGRL 模型中有 3 个部分使用了超参数, 分别是池化率(即每层池化保留多少比例的节点)、解释的层数 l, 以及解释损失函数权重 $\lambda_1, \lambda_2, \cdots, \lambda_l$. 此外, 节点特征编码器的选择也会对实验结果准确率产生影响. 接下来我们将分别围绕这 3 类参数进行实验并分析其实验结果.

在池化操作中, 我们由低到高选择了池化率: 0.10, 0.25, 0.50, 0.75, 0.90, 并测试图分类准确率. 池化率对于图分类结果的影响如图 4 所示, 可以得出模型在 MUTAG, PROTEINS, NCI1 数据集上的池化率处于 0.75 时效果最好, 在池化率更高时, 由于保留的信息过多, 其准确率略有下降. 只有在 IMDB-BINARY 数据集上池化率为 0.9 时, 分类准确率最高, 其原因主要是 IMDB-BINARY 数据集中结构信息更为重要, 因

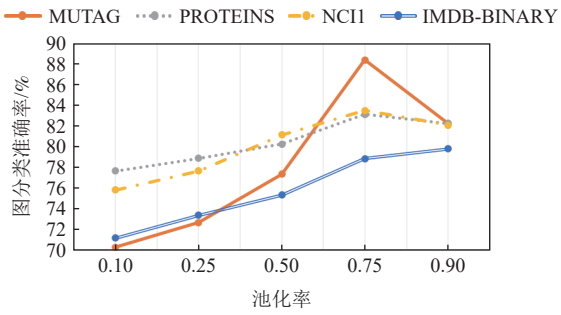


Fig. 4 The impact of pooling ratio on graph classification results

图 4 池化率对图分类结果影响

此保留的节点更多, 对于分类结果的准确率贡献更高. 而当池化率过低时, 由于 HSEGRL 模型保留的准确率信息变少, 其准确率大幅度下降, 说明了解释池化的重要性及有效性.

此外, 我们还探讨了解释子损失函数权重对结果的影响, 为了更好地展示解释权重之间的关系, 我们以 2 层解释为例, 在 MUTAG 数据集上测试了不同层解释权重对结果的影响. λ_1, λ_2分别表示 1 层及 2 层解释子解释权重, 图分类任务损失函数 $\mathcal{L}_{GC}$ 的权重则为 $1-\lambda_1-\lambda_2$. 我们分别对 λ_1, λ_2取值 0.1~0.9, 测试 2 层解释的图分类准确率. 同时, 为确保图分类任务权重小于等于 1. 测试结果如图 5 所示.

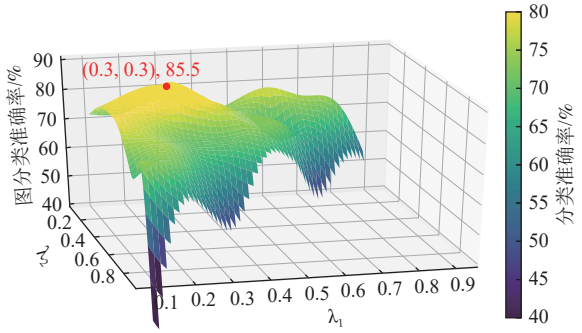


Fig. 5 The impact of interpreter's loss function weights on graph classification accuracy

图 5 解释子损失函数权重对图分类准确率的影响

由图 5 可得, 采用 2 层解释子的图分类准确率最高为 85.55%, 此时解释权重 λ_1和 λ_2均为 0.3, 与基于 3 层解释子的 HSEGRL 模型(模型准确率 88.42%, 见 4.2 小节)相比, 其准确率下降 2.87 个百分点. 而 λ_1, λ_2 取值较小时, 解释损失占比较低, 模型进行图分类时对于层次解释信息考虑较少, 因而图分类准确率会有所下降; 而对于某一层解释信息关注较少的情况下, 即 λ_1=0.1 或 λ_2=0.1 时, 其准确率相较于该层次解释信息被充分考虑情况下略低; 随着 λ_1与 λ_2占比增大, 虽然 HSEGRL 模型获取到了更多的解释信息, 但由于图分类损失函数占比降低, 导致了图分类准确率大幅度下降. 通过以上分析可知, HSEGRL 模型只有综合考虑每层解释所获取的信息及图分类任务的信息, 才能在提高图分类准确率的同时达到较好的解释效果.

对于解释层数的有效性, 我们设计了解释子层数对于准确率的影响实验, 其中为了确保解释的贡献一致, 我们将每层解释的损失权重设为 0.2, 其结果如表 3 所示.

Table 3 The Influence of the Interpreter Layer's Number on Graph Classification Accuracy

表 3 解释子层数对图分类准确率的影响 %

l	MUTAG	PROTEINS	NCI1	IMDB-BINARY
1	78.33±7.25	77.75±6.67	62.80±2.25	65.51±1.30
2	82.23±7.51	70.27±1.17	78.25±1.23	**78.85±1.23**
3	**88.42±5.31**	**77.15±2.27**	**83.53±2.03**	77.85±1.15
4	66.67±4.45	72.97±1.87	73.25±2.25	72.25±3.31
5	50.21±3.31	56.65±1.13	62.12±2.21	56.56±1.13

注: 黑体数值表示最优, "±"前后数值表示平均准确率和准确率误差.

由表 3 可知, 在分子与化学数据集中, 解释层数在 3 层时能取得最好效果, 而在 2 层解释层数时同样也能取得较好的解释效果. 而随着解释层数增多, 图分类损失函数占比降低, 模型发生过拟合, 导致分类准确率降低. 而在社交网络数据集 IMDB-BINARY 中由于 2 层解释能更好地表征其网络结构信息, 因此 2 层解释能取得更好的结果. 由于在大多数数据集中, 3 层解释都能取得较好效果, HSEGRL 模型选择 3 层解释进行实验, 而对于一些特殊数据集可以通过调整解释层数调整模型解释层次信息使模型取得更好的分类准确率.

GNN 编码器的选择同样会对实验结果产生影响. 针对 MUTAG, PROTEINS, NCI1, IMDB-BINARY 数据集, 选择了经典的 GNN 模型作为编码器, 包括: GCN, GAT, GraphSAGE, Chebnet 测试其图分类准确率, 其中解释层数为 3 层, 池化率选择 0.75, 每层解释损失函数权重为 0.2.

其结果如图 6 所示, 不同的编码器对于不同的数据集影响不同, 总体而言, GAT 效果较好, 而 GraphSAGE 效果较差. GAT 编码器为 HSEGRL 模型引入了注意力机制, 能够从一定程度上表征图拓扑信息, 从而提升 GNN 模型的精度, 因此能够提升图分类的准确率. 而使用 GraphSAGE 作为编码器的 HSEGRL

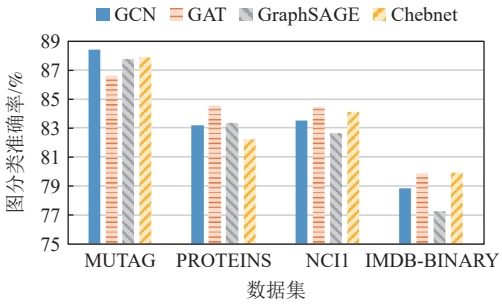


Fig. 6 The impact of encoder on the accuracy of graph classification

图 6 编码器对于图分类准确率的影响

由于引入采样的方法导致损失了部分精度，然而GraphSAGE由于其对于大规模图的处理能力，使得HSEGRL能够扩展至大规模数据集. 而在MUTAG数据集上，使用了GCN作为编码器的HSEGRL，其分类准确率比使用GAT作为编码器的模型高，由于MUTAG数据集单张图规模较小，其注意力系数对结果的影响较小，故而使用GCN作为编码器能取得更好的效果. 基于以上分析可得：不同的编码器会根据其本身具有的优势，对于HSEGRL产生影响，需要根据不同的应用场景选择不同的编码器，以提升HSEGRL的精度及可扩展性.

综合上述讨论可知，池化率、解释子层数以及解释子损失函数的权重对模型的分类准确率有显著影响，表明模型参数具有较高的敏感性. 然而，实验结果表明，在大多数数据集上，当池化率设定在0.75~0.9时，模型能够取得最优的实验结果. 此外，大多数模型在使用3层解释子且每层损失函数值设置为0.2时表现最佳. 对于包含更多层次信息的大规模数据集，考虑采用4层解释子并相应调整每层损失函数的权重可能会有更好的效果.

3.5 模型时间复杂度分析

根据算法1，HSEGRL在时间开销上有所增加，主要体现在编码、节点重要性评估和解码器3个环节. 与此同时，子图与特征重构的方法基于矩阵运算，不会增加额外的时间复杂度. 理论上，大多数MPNN编码器的时间复杂度为$O\left(L\|\boldsymbol{A}\|_0 d+Lnd^2\right)$，其中$L$是层数，$n$是节点数，$\|\boldsymbol{A}\|_0$是邻接矩阵$\boldsymbol{A}$中的非零元素的数量，$d$是节点特征维度. MLP的时间复杂度仅与图特征相关，不涉及邻居节点的访问，时间复杂度可以简化为$O(1)$. 据此，整个模型的时间复杂度可以近似为$O(3L\|\boldsymbol{A}\|_0 d+3Lnd^2)$.

我们在数据集MCF-7, SW-620, YeastH[42]上设计了时间效率实验，其具体参数见表1. 我们分别测试了最基础的GCN模型、仅完成图表示学习任务的HoscPool模型和HSEGRL模型的运行时间与准确率，其中运行时间以s/epoch表征. 其中，模型参数设置与实验4.1小节保持一致. 结果如表4所示.

根据表4，当数据集规模较小时，HSEGRL和GCN模型每个epoch的时间差异不大. 随着数据集规模的增大，HSEGRL的运行效率虽有所下降，但相较于仅专注于图表示学习的模型HoscPool，其效率仍然更高. 而对比自解释模型ProtGNN，HSEGRL由于其需要学习自身每层的决策模式会产生额外的开销，HSEGRL的效率略低于ProtGNN. 考虑到HSEGRL在完成图表示学习的同时具备分层自解释能力，额外的时间开销是可以被接受的.

Table 4 Graph Classification Experimental Results of Large Datasets

表4 大数据集图分类实验结果

模型	参数	MUTAG	MCF-7	SW-620	YeastH
GCN	运行时间/s	1.85	6.41	8.37	15.89
	准确率/%	72.22	91.11	93.73	87.20
HoscPool	运行时间/s	2.45	128.25	133.87	172.94
	准确率/%	87.13	91.27	94.12	87.46
ProtGNN	运行时间/s	2.12	11.46	15.28	31.53
	准确率/%	80.50	91.03	93.14	87.15
HSEGRL	运行时间/s	2.39	14.71	17.52	35.25
	准确率/%	88.42	92.21	94.40	88.13

4 总　　结

在本文中，我们首先总结了现有可解释图表征学习的主要方法，并分析了现有工作在分层解释领域缺乏的问题. 基于现有解释工作在层次信息利用上有所欠缺的问题，本文提出了一种可以分层自解释的图解释模型，通过获取图层次信息，对图表示学习模型进行自解释. 首先，提出了一个创新的分层自解释模型HSEGRL的框架，随后，创新性地提出了解释子概念，在优化解释信息的同时，使用提取到的解释信息优化预测结果.然后，本文通过设计可自解释的EAPooling方法，同时评估图的节点特征与拓扑特征并自适应调整其权重，实现可解释子图的提取. 为了证明本文方法的有效性，我们设计了一系列实验，在真实世界数据集与人工合成数据集的图分类任务中通过准确率验证预测的有效性，同时通过可视化的方法验证分层解释的效果，最后，我们通过参数分析，说明了模型参数对模型的预测结果产生的影响.

在未来，本文工作还具有扩展至节点分类领域的潜力，该工作在节点分类领域主要面临着数据规模较大的问题. 此外，还能通过加入高阶图结构信息，如图核、motif等，实现高阶图拓扑的分层解释.

作者贡献声明：李平提出论文的研究框架，并进行论文撰写；宋舒寒协助进行部分实验；张园负责优化论文写作语法；曹华伟负责把控科研进度；叶笑春负责科研思路指导；唐志敏负责把控论文整体架构.

参 考 文 献

[1] Yang Donghua, He Wang, Wang Jinbao. Survey on knowledge graph embedding learning[J]. Journal of Software, 2022, 33(9): 3370−3390 (in Chinese)
(杨东华, 何 王, 王金宝. 面向知识图谱的图嵌入学习研究进展[J]. 软件学报, 2022, 33(9): 3370−3390)

[2] Zhang Xiongtao, Zhu Na, Guo Yuhui. A survey on session-based recommendation methods with graph neural network [J]. Data Analysis and Knowledge Discovery, 2024, 8(2): 1−20 (in Chinese)
(张雄涛, 祝娜, 郭玉慧. 基于图神经网络的会话推荐方法综述 [J]. 数据分析与知识发现, 2024, 8(2): 1−20)

[3] Zhang Wei, Li Zhang, Wang Jianyong. A user trajectory identification model with fusion of spatio-temporal behavior and social relation[J]. Chinese Journal of Computers, 2021, 44(11): 2173−2188 (in Chinese)
(张伟, 李 张, 王建勇. 融合时空行为与社交关系的用户轨迹识别模型[J]. 计算机学报, 2021, 44(11): 2173−2188)

[4] Chen Fenxiao, Wang Y, Wang Bin, et al. Graph representation learning: A survey[J]. APSIPA Transactions on Signal and Information Processing, 2020, e15((9):): 1−21

[5] Xu Bingbing, Cen Keyan, Huang Junjie, et al. A survey on graph neural network[J]. Chinese Journal of Computers, 2020, 43(5): 755−780
(徐冰冰, 岑科廷, 黄俊杰, 等. 图卷积神经网络综述[J]. 计算机学报, 2020, 43(5): 755−780)

[6] Ma Shuai, Liu Jianwei, Zuo Xin. Survey on graph convolution neural network[J]. Journal of Computer Research and Development, 2022, 59(1): 47−80 (in Chinese)
(马帅, 刘建伟, 左信. 图神经网络综述[J]. 计算机研究与发展, 2022, 59(1): 47−80)

[7] Kakkad J, Jannu J, Sharma K, et al. A survey on explainability of graph neural networks [J]. arXiv preprint, arXiv: 2306.01958, 2023

[8] Ying Zhitao, Bourgeois D, You Jiaxuan, et al. GNNExplainer: Generating explanations for graph neural networks[C]//Proc of the 33rd Int Conf on Neural Information Processing Systems. New York: Curran Associates Inc, 2019: 9244–9255

[9] Luo Dongsheng, Cheng Wei, Xu Dongkuan, et al. Parameterized explainer for graph neural network[J]. Advances in Neural Information Processing Systems, 2020, 33(1): 19620−19631

[10] Dai Enyan, Wang Suhang. Towards self-explainable graph neural network[C]//Proc of the 30th ACM Int Conf on Information & Knowledge Management. New York: ACM, 2021: 302−311

[11] Sui Yongduo, Wang Xiang, Wu Jiancan, et al. Causal attention for interpretable and generalizable graph classification[C]//Proc of the 28th ACM SIGKDD Conf on Knowledge Discovery and Data Mining. New York: ACM, 2022: 1696−1705

[12] Lin Wanyu, Lan Hao, Wang Hao, et al. Orphicx: A causality-inspired latent variable model for interpreting graph neural networks[C]//Proc of the IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2022: 13729−13738

[13] Lin Wanyu, Lan Hao, Li Baochun. Generative causal explanations for graph neural networks[C]//Proc of Int Conf on Machine Learning. New York: PMLR, 2021: 6666−6679

[14] Pope P E, Kolouri S, Rostami M, et al. Explainability methods for graph convolutional neural networks[C]//Proc of The IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 10772−10781

[15] Huang Qiang, Yamada M, Tian Yuan, et al. GraphLime: Local interpretable model explanations for graph neural networks[J]. IEEE Transactions on Knowledge and Data Engineering, 2022, 35(7): 6968−6972

[16] Feng Aosong, You Chenyu, Wang Shiqiang, et al. KerGNNs: Interpretable graph neural networks with graph kernels[C]//Proc of the AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2022: 6614−6622

[17] Han Xuanyuan, Barbiero P, Georgiev D, et al. Global concept-based interpretability for graph neural networks via neuron analysis[C]// Proc of the AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2023: 10675−10683

[18] Khoshraftar S, An Aijun. A survey on graph representation learning methods[J]. ACM Transactions on Intelligent Systems and Technology, 2024, 15(1): 1−55

[19] Gao Hongyang, Ji Shuiwang. Graph U-nets[C]//Proc of Int Conf on Machine Learning. New York: PMLR, 2019: 2083−2092

[20] Velickovic P, Cucurull G, Casanova A, et al. Graph attention networks[J]. Stat, 2017, 1050(20): 48510−48550

[21] Cangea C, Veličković P, Jovanović N, et al. Towards sparse hierarchical graph classifiers [J]. arXiv preprint, arXiv: 1811.01287, 2018

[22] Grattarola D, Zambon D, Bianchi F, et al. Understanding pooling in graph neural networks[J]. IEEE Transactions on Neural Networks and Learning Systems, 2022, 35(2): 2708−2718

[23] Ying Zhitao, You Jiaxuan, Morris C, et al. Hierarchical graph representation learning with differentiable pooling[C]//Advances in Neural Information Peocessing Systems. La Jolla, California: Curran Associates Inc, 2018, 1−10

[24] Lee J, Lee I, Kang J. Self-attention graph pooling[C]//Proc of Int Conf on Machine Learning. New York: PMLR, 2019: 3734−3743

[25] Tang Haoteng, Ma Guixiang, He Lifang, et al. CommPool: An interpretable graph pooling framework for hierarchical graph representation learning[J]. Neural Networks, 2021, 143(1): 669−677

[26] Gilmer J, Schoenholz S S, Riley P F, et al. Neural message passing for Muaum chemistry[C]// Proc of Int Conf on Machine Learning. New York: PMLR, 2017: 1263−1272

[27] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[C]//Proc of Int Conf on Learning Representations.2016[2024−05−18].https://openreview.net/forum?id=SJU4ayYgl

[28] Xu Keyulu, Hu Weihua, Leskovec J, et al. How powerful are graph neural networks?[C]//Proc of Int Conf on Learning Representations. 2019 [2024−05−18].https://openreview.net/forum?id=ryGs6iA5Km

[29] Duval A, Malliaros F. Higher-order clustering and pooling for graph neural networks[C]//Proc of the 31st ACM Int Conf on Information & Knowledge Management. New York: ACM, 2022: 426−435

[30] Debnath A K, Lopez de Compadre R L, Debnath G, et al. Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity[J]. Journal of Medicinal Chemistry, 1991, 34(2): 786−797

[31] Wale N, Watson I A, Karypis G. Comparison of descriptor spaces for

chemical compound retrieval and classification[J]. Knowledge and Information Systems, 2008, 14(1): 347−375

[32] Dobson P D, Doig A J. Distinguishing enzyme structures from non-enzymes without alignments[J]. Journal of Molecular Biology, 2003, 330(4): 771−83

[33] Borgwardt K M, Ong C S, Schnauer S, et al. Protein function prediction via graph kernels[J]. Bioinformatics, 2005, 21(suppl_1): i47−i56

[34] Yanardag P, Vishwanathan S V N. Deep graph kernels[C]//Proc of the 21st ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2015: 1365−1374

[35] Zhang Muhan, Cui Zhicheng, Neumann M, et al. An end-to-end deep learning architecture for graph classification[C]//Proc of the AAAI Conf on Artificial Intelligence. Palo Alto: Assoc Advancement Artificial Intelligence. Palo Alto, CA: AAAI, 2018: 32−40

[36] Bouritsas G, Frasca F, Zafeiriou S, Bronstein M M. Improving graph neural network expressivity via subgraph isomorphism counting[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 45(1): 657−668

[37] Zhao Lingxiao, Jin Wei, Akoglu L, et al. From stars to subgraphs: Uplifting any GNN with local structure awareness [J]. arXiv preprint, arXiv: 211003753, 2021

[38] Zhang Zaixi, Liu Qi, Wang Hao, et al. ProtGNN: Towards self-explaining graph neural networks[C]//Proc of the AAAI Conf on Artificial Intelligence. Palo Alto, AC: AAAI, 2022: 9127−9135

[39] Yu Junchi, Cao Jie, He Ran. Improving subgraph recognition with variational graph information bottleneck[C]//Proc of the IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2022: 19396−19405

[40] Yu Junchi, Xu Tingyang, Rong Yu, et al. Graph information bottleneck for subgraph recognition [J]. arXiv preprint, arXiv: 201005563, 2020

[41] Yuan Hao, Yu Haiyang, Gui Shurui, et al. Explainability in graph neural networks: A taxonomic survey[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 45(5): 5782−99

[42] Yan Xifeng, Cheng Hong, Han Jiawei, et al. Mining significant graph patterns by leap search[C]//Proc of the 2008 ACM SIGMOD Int Conf on Management of Data. New York: ACM, 2008: 433−444



Li Ping, born in 1995. PhD candidate. Student member of CCF. Her main research interest includes the optimization of graph neural network models and the explainability.

李　平, 1995 年生. 博士研究生. CCF 学生会员. 主要研究方向为图神经网络模型优化与可解释性.



Song Shuhan, born in 1997. PhD candidate. Student member of CCF. His main research interests include graph neural networks, graph self-supervised learning, and graph neural network architecture search and robustness on graph learning. (songshuhan19s@ict.ac.cn)

宋舒寒, 1997 年生. 博士研究生. CCF 学生会员. 主要研究方向为图神经网络、图自监督学习、图神经网络架构搜索及图学习鲁棒性.



Zhang Yuan, born in 1990. PhD candidate. Member of CCF. Her main research interests include graph processing and parallel computing. (zhangyuan-ams@ict.ac.cn)

张　园, 1990 年生. 博士研究生. CCF 会员. 主要研究方向为图处理与并行计算.



Cao Huawei, born in 1989. PhD, associate professor. Member of CCF. His main research interests include parallel computing and high throughput computing architecture.

曹华伟, 1989 年生. 博士, 副研究员. CCF 会员. 主要研究方向为并行计算、高通量计算架构.



Ye Xiaochun, born in 1981. PhD, professor. Member of CCF. His main research interests include algorithm paralleling and optimizing, software simulation, and architecture for high throughput computer. (yexiaochun@ict.ac.cn)

叶笑春, 1981 年生. 博士, 研究员. CCF 会员. 主要研究方向为算法并行和优化、软件仿真、高通量计算机架构.



Tang Zhimin, born in 1966. PhD, professor. Senior member of CCF. His main research interests include high performance computer architecture, processor design, and digital signal processing. (tang@ict.ac.cn)

唐志敏, 1966 年生. 博士, 研究员. CCF 高级会员. 主要研究方向为高性能计算机体系结构、处理器设计、数字信号处理.