

基于图神经网络的多维度池化图分类^①



王淑栋, 安 迪, 庞善臣

(中国石油大学(华东) 计算机科学与技术学院, 青岛 266580)

通信作者: 王淑栋, E-mail: shudongwang2013@SOHU.COM

摘 要: 目前很多处理图数据的图神经网络方法被提出, 然而大多数研究侧重于对特征聚合的卷积层的研究而不是进行下采样的池化层. 此外, 形成聚类簇的池化方式需要额外计算分配矩阵; 节点得分的池化方式排名方式单一. 为解决上述问题, 提高图分类任务的准确性, 本文提出了一种新的基于多维度信息的图池化算子 MDPool. 该模型使用节点特征信息以及图拓扑结构信息, 获取不同维度下的节点得分. 使用注意力机制归纳不同维度下的得分权重, 生成更为健壮的节点排名, 基于节点排名自适应选择节点集合生成诱导子图. 提出的 MDPool 可以集成到多种的图神经网络结构, 将 MDPool 池化算子与图神经网络卷积层堆叠形成编码解码模型 EDMDPool. 在 4 个公开数据集的图分类任务中, EDMDPool 均高于现有基线模型.

关键词: 图池化; 图分类; 图神经网络; 多头自注意力; 中心性

引用格式: 王淑栋, 安迪, 庞善臣. 基于图神经网络的多维度池化图分类. 计算机系统应用, 2023, 32(6): 22-31. <http://www.c-s-a.org.cn/1003-3254/9118.html>

Multi-dimensional Graph Pooling for Graph Classification Based on Graph Neural Networks

WANG Shu-Dong, AN Di, PANG Shan-Chen

(College of Computer Science and Technology, China University of Petroleum, Qingdao 266580, China)

Abstract: For processing graph data, a variety of graph neural network approaches have been developed; however, most research focuses on the convolutional layer for feature aggregation rather than the pooling layer for downsampling. Additionally, the computation of assignment matrices is required for the pooling approach to creating clusters, and the pooling method for node scores simply employs one scoring strategy. A new multi-dimension graph pooling operator, MDPool, is presented to solve these issues and increase the precision of graph classification tasks. To calculate node scores in various dimensions, the model makes use of information on node features and graph structure. The score weighting across several dimensions is summarized by using an attention technique to provide more reliable node rankings. The set of nodes is chosen to produce induced subgraphs based on the node rankings. The proposed MDPool can be implemented into a variety of graph neural network architectures. The encode-decode model, EDMDPool, is created by stacking the MDPool pooling operator with the convolutional layer of the graph neural network. In the graph classification tasks of four public datasets, EDMDPool performs better than the existing baseline model.

Key words: graph pooling; graph classification; graph neural networks (GNN); multi-head self-attention; centrality

深度学习的出现, 使得识别和使用数据的技术, 得到了发展^[1]. 特别是卷积神经网络 (CNN), 在图像处理领域^[2,3] 取得了巨大的成功. 然而, 现实世界还存在许

多不包含位置信息和序列信息的数据, 如生物网络^[4]、化合物分子网络^[5]、社会网络^[6] 等数据, 此类数据可以由图的节点及边构成. 借鉴 CNN 的卷积层池化层的功

① 基金项目: 国家自然科学基金 (61873281)

收稿时间: 2022-11-30; 修改时间: 2022-12-23; 采用时间: 2023-01-06; csa 在线出版时间: 2023-03-24

CNKI 网络首发时间: 2023-03-27

能,在图数据上推广卷积和池化操作是相当有吸引力的^[7].将CNN中的卷积操作扩展到任意图结构数据上的方法,称为图神经网络(GNN).目前各种GNN算法在以节点为主的图学习任务中获得出色的表现,如节点分类^[8,9],链路预测^[10],推荐系统^[11]等.图分类任务作为一种重要的数据挖掘任务,在很多领域均有应用前景^[12].例如,在生物信息学中,对蛋白质网络分类,判断该蛋白是否为酶,对某种疾病是否具有治疗能力^[13,14];在化学信息学中,对化合物分子图进行分类,判断分子的诱变性、抗癌活性、毒性^[15,16];给定分子的结构图,判断其性质^[17].图中的池化操作在图的分层表示中起着关键作用,可以挖掘图中存在的子图结构,获取全局特征的层次信息^[18].针对图中池化操作进行设计,对解决图分类问题具有较高的研究价值.

图分类任务的目的是根据给定图的节点特征以及拓扑结构来预测整个图的标签.早期的图分类学习模型主要是通过设计全局特征提取函数来获取所有节点的特征.虽然这种方式是可行的,但是通过这种方式得到的图级别表示本质上是“扁平的”,只考虑了图的特征信息,图的层次结构信息在一定程度上被忽略了^[19].为了捕捉层次结构信息用于图分类,最近一些工作根据图的不同子结构和节点的不同作用,专注于GNN的分层池化设计^[18,20-22].主要有两类方法,一种是将节点分组形成节点簇,另一种是选取代表节点形成子图.在节点簇的池化方法中,使用学习的方式得到节点分配矩阵,根据分配矩阵将每个节点进行分簇,然后将簇内节点汇集成新节点,形成粗化图^[18,21].由于此类方法需要额外学习分配矩阵,导致模型的计算复杂度较高.在选取代表节点的方法中,核心思想是计算出图中节点的排名,对原始图进行节点抽取,形成粗化图^[20,22].例如可学习的投影向量^[20],邻域重建中心节点得分^[23]等.由于节点排名时没有显式的使用图的邻接矩阵信息,并且对节点的排名方式单一,观点往往具有偏见^[24].不同维度的信息对节点评分侧重点可能不同,因此在多个维度信息下协作学习稳健的节点排名是非常具有必要性的^[25].

为了解决上述问题的局限性,我们提出了一个新的基于编码解码结构的多维度图池化方法EDMDPool.它具有编码解码框架,可以学习图的分层表示.具体来说我们首先引入了一套度量方法,来评估特征维度和拓扑结构维度下节点的两种重要性分数.然后,通过注意力机制^[26],动态地将两种维度下的重要性分数进行

汇总,来生成更为稳健的节点排名.之后,编码结构中的池化操作根据得到的节点排名自适应的选择排名前百分之 K 的节点集合,生成原始图的诱导子图,并记录所选节点的编号,用于解码部分恢复原始结构.我们堆叠GNN层和MDPool层来实现一个端到端的架构,以执行图分类任务.

1 相关工作

与CNN中池化效果相似,GNN池化可以使参数减少,得到更好的泛化性能^[22].目前图池化的方法可以分为两类:全局图池化和分层图池化.

全局图池化:全局池化的方法使用平均操作、加和操作或神经网络将所有节点嵌入整合为特征向量作为图的表示^[22].Set2Set使用LSTMs模型寻找图中重要节点,聚合节点信息生成图表示^[27].DGCNN根据顶点在图结构中的作用,对节点嵌入进行排序后传递给传统CNN结构对图进行表示^[28].全局池化虽然可以应用于不同大小的图,具有不同结构的图可以被统一处理,但是这种方法只根据节点特征进行操作,可能会丢失一部分结构信息.

分层图池化:分层表示对于获取图的层次结构信息至关重要,可以学习每层基于特征或拓扑的节点分配.在节点分组形成节点簇的方法中,Ying等人^[18]提出DiffPool,一种可微分池化模型,使用GNN将节点分配到可重复的节点集合中.Diehl等人^[29]提出了EdgePool,通过学习节点之间边的得分决定节点是否融合,这种方式导致了每次池化节点数减半.在选取代表节点形成子图的方法中,顾昕等人^[30]提出节点信息保留的多头注意力图池化模型MHAPool,与本文提出的特征维度下多头自注意力分数相似,但本文包括了拓扑维度下节点排名信息的融合.张红梅等人^[31]提出层内-层间联合特征提取的模型.朱小草等人^[32]提出一种基于稀疏注意力自适应图池化方法AGP-SA,保留图的局部特征.Lee等人^[22]提出了SAGPool,它是一种用GNN获取自注意力得分的图池化方法.Gao等人提出了具有池化和反池化操作的Graph U-Net,池化层根据节点在可训练投影向量 p 上的得分排名,选择节点形成新子图^[20].在Ranjan等人^[33]的工作中,利用新的自注意网络和改进的GNN来捕捉给定图中每个节点的重要性,设计了ASAPool.Zhang等人^[34]提出了图自适应池化方法GSAPool,根据节点的局部结构和特征信

息以多种方式评估节点的重要性. 然而现有的大多数方法, 有限的使用了图的特征信息或拓扑结构, 忽略了不同维度数据协作贡献.

2 多维度池化图分类模型

2.1 问题定义

图分类问题的任务是根据给定信息找到图对应标签的映射关系. 具体来说, 给定一数据集 $D = \{(G_1, Y_1), (G_2, Y_2), (G_3, Y_3), \dots\}$, 获取映射函数 $F: G \rightarrow Y$, 其中 $G = \{G_1, G_2, \dots, G_n\}$ 是输入图的集合, $Y = \{Y_1, Y_2, \dots, Y_n\}$ 是与之相对应的标签集合. 对于数据集中任意一个图 $G = (V, E, X)$, 标签为 Y , V 表示节点集合, 节点数量为 n , E 表示图中边的集合, 边的数量为 e . 我们令 X 作为图的特征矩阵, $X \in \mathbb{R}^{n \times d}$, 其中 d 为节点的特征维度. 此外, 令 $A \in \mathbb{R}^{n \times n}$ 作为图的邻接矩阵, 表明图中节点的连接关系, $D \in \mathbb{R}^{n \times n}$ 是邻接矩阵 A 的度矩阵.

2.2 模型框架

全局多维度池化图分类框架流程: 根据文献 [28] 提出的全局池化架构, 本文设计了相似的结构, 如图 1(a) 所示, 全局多维度池化框架由 3 层图卷积层堆叠, 并将每层的特征进行串联拼接后传入多维度图池化层, 经读出层获取到图的特征表示, 最后经过线性层进行分类.

编码-解码多维度池化图分类框架流程: 受文献 [20,35] 中编码器和解码器结构的启发, 本文设计了基于编码-解码多维度池化的图分类框架. 如图 1(b) 所示, 堆叠编码块来构建编码器部分, 每个编码块包含一个 GCN 层和一个 MDPool 层, GCN 层负责聚合一阶邻居节点信息, MDPool 层负责缩减图的规模. 在解码器部分, 设计与编码器相同数目的解码块, 每个解码块包含一个 UnPool 层和一个 GCN 层, UnPool 层负责恢复对应编码部分的图结构, GCN 层负责聚合信息. 将编码块、解码块中的不同层次特征进行读出, 图的特征使用读出层输出进行拼接, 最后将图的特征送入线性层进行分类. 图 2 给出了由两个编码块两个解码块组成的多维度池化图分类框架的解释. 对于输入图数据, 每层之间使用 GCN 提取节点特征. 在编码部分, 使用 MDPool 缩减原始图节点数目, 生成诱导子图, 并记录节点下标及图结构用于解码部分恢复. 在解码部分, UnPool 层使用记录的结构及下标恢复原始图结构. 对编码解码过程中每层子图特征进行读出, 将拼接后的特征传入 MLP 获取标签, 用于图分类.

在本文提出的模型框架中, 我们设计了基于节点特征维度和图拓扑结构维度协作的多维度图池化层, 为了对比单个维度下的池化层的效果, 我们设计仅包含特征维度的池化层模型和仅包含拓扑结构维度的池化层模型做验证. 我们将在第 3.4 节实验部分讨论该问题.

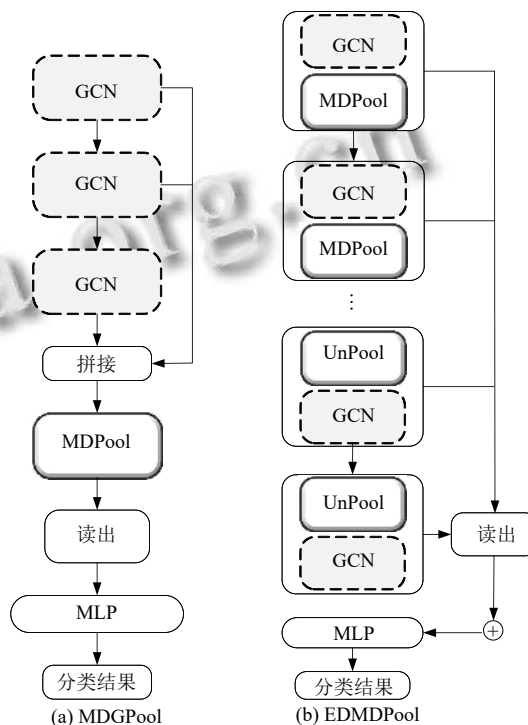


图 1 模型框架流程

2.3 多维度图池化层

在本节中, 我们将介绍提出的多维度图池化操作 MDPool, 以实现对比图的下采样. 根据文献 [20,22] 中图池化方式的启发, 图池化操作用来确定诱导子图的节点集, 节点集的采样根据原始节点重要性排名前百分之 K 的节点形成. 在这里, 我们设计了一个新的多维度图池化操作, 可以使不同维度下的节点得分进行协作, 从而产生更健壮的节点排名.

基于节点排名的图池化操作的关键是获得图中所有节点重要性排名的排序. 一般来说, 图通常包含结构信息和节点特征信息, 这两类信息对应着图的邻接矩阵 A 和特征矩阵 X . 因此, 根据这两类信息, 评估每个节点基于特征和结构维度的排名. 图 3 给出了我们所提出的图池化层的说明解释. 假设在模型框架的第 l 层中, 邻接矩阵 $A^l \in \mathbb{R}^{N^l \times N^l}$, 其中 N^l 为第 l 层中节点数目. 特征矩阵 $X^l \in \mathbb{R}^{N^l \times d}$, 第 l 层中每个节点包含 d 个特征. MDPool 的具体操作如下.

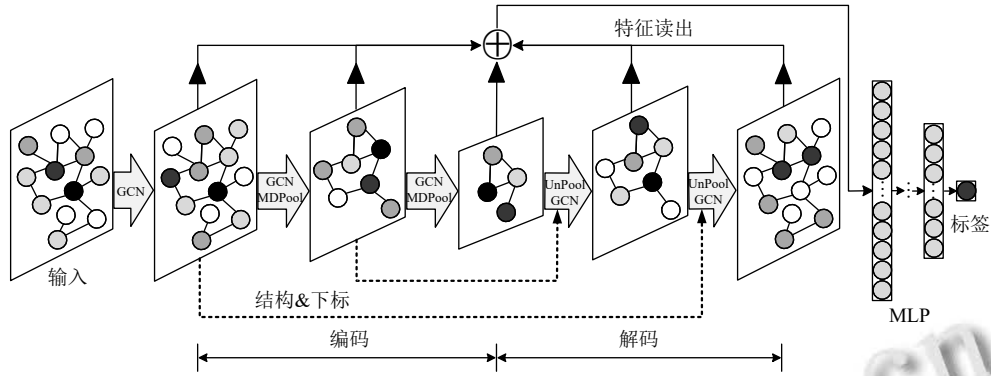


图2 两层编码解码多维度图分类模型框架

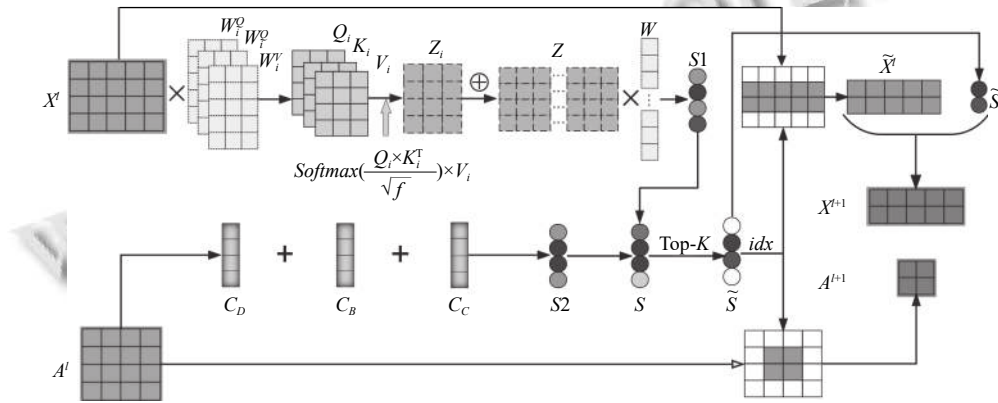


图3 MDPool层的说明

节点特征维度: 图通常包含丰富的特征信息, 节点特征在很大程度上可以描述自身属性. 我们使用节点特征的多头自注意力^[36]分数, 作为图中节点在特征维度的得分. 将节点特征 X^l 做 3 次线性映射得到 Q_i, K_i, V_i , 如下所示:

$$\begin{cases} Q_i = X^l W_i^Q \\ K_i = X^l W_i^K \\ V_i = X^l W_i^V \end{cases} \quad (1)$$

其中, $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d \times f}$ 为可学习的参数矩阵, $Q_i, K_i, V_i \in \mathbb{R}^{N^l \times f}$ 代表多头自注意力中的第 i 个头的 query、key、value, $i \in \{1, 2, \dots, h\}$, h 为多头自注意力中头的数目.

Q_i, K_i, V_i 中的每一行表示为每个节点不同空间下的特征向量, 则经过式 (2) 得到注意力矩阵, 并将注意力矩阵应用于特征矩阵 V_i , 过程如下所示:

$$O_i = \text{Softmax}\left(\frac{Q_i K_i^T}{\sqrt{f}}\right) \quad (2)$$

$$Z_i = O_i V_i \quad (3)$$

其中, $O_i \in \mathbb{R}^{N^l \times N^l}$ 表示为节点之间的注意力或者相关

性, $\text{Softmax}(\cdot)$ 函数使 O_i 每一行元素相加为一. 将节点特征 V_i 与注意力矩阵进行加权线性组合, 得到图在第 i 个头下的特征矩阵 $Z_i \in \mathbb{R}^{N^l \times f}$, 使每个节点特征向量都含有当前图中所有节点特征向量的信息.

将每个注意力头中获取到的节点特征向量进行拼接, 得到在多头自注意力机制下的特征向量 $Z \in \mathbb{R}^{N^l \times (h \times f)}$. 为了得到所有节点的排名, 我们将多头特征向量进行线性变换, 得到一维表示 $S_1 \in \mathbb{R}^{N^l \times 1}$:

$$Z = \text{concat}(Z_1, \dots, Z_h) \quad (4)$$

$$S_1 = \sigma(ZW) \quad (5)$$

其中, $W \in \mathbb{R}^{(h \times f) \times 1}$ 为可训练的矩阵, $\sigma(\cdot)$ 为 *Sigmoid* 非线性激活函数, 将节点得分范围限制在 $[0, 1]$.

图拓扑结构维度: 除了特征信息, 节点的重要性通常还与图拓扑结构相关. 图拓扑结构本身可以提供许多有用信息, 比如节点的度以及节点之间的最短路径等. 因此, 在这里我们使用节点的度中心性、介数中心性^[37]、接近中心性^[38] 作为节点在图拓扑结构维度下得分. 我们认为使用单一中心性研究节点的重要程度具有偏差, 使用 3 种中心性结合不同角度信息计算节点

重要性更为准确。

一个节点与其他节点关系越多,该节点的影响力就越强,因此,度中心性越大的节点其重要性越大.度中心性描述如下所示:

$$C_D = \frac{\text{degree}(A^l)}{N-1} \quad (6)$$

其中, $C_D \in \mathbb{R}^{N \times 1}$, $\text{degree}(\cdot)$ 函数通过计算邻接矩阵中每个节点的连接数从而得到节点的度.由于只关注了与某一个节点直接相连的点数,忽略间接相连的点数,因此被视为局部中心度.

图中两个不相邻节点的相互作用取决于其他节点,特别是处于两节点最短路径上的节点.因此,一节点处于多条其他节点的最短路径中,那么该节点就是图中重要节点,有较大的中介中心性,其描述如下:

$$C_B(v) = \sum_{s,t \in V} \frac{f(s,t|v)}{f(s,t)} \quad (7)$$

其中, $C_B \in \mathbb{R}^{N \times 1}$ 表示节点的介数中心性, V 是节点集合, $f(s,t)$ 表示节点 s 到节点 t 最短路径的数目, $f(s,t|v)$ 表示节点 s 到节点 t 最短路径上包含节点 v 的路径数目.

接近中心性是一节点与图中其他节点的最短距离之和的倒数,代表了该节点与其他点的远近程度,其描述如下:

$$C_C(v) = \frac{n-1}{\sum_{u \in \{V-v\}} d(v,u)} \quad (8)$$

其中, $C_C \in \mathbb{R}^{N \times 1}$ 表示节点的接近中心性, $d(v,u)$ 是节点 v 和节点 u 之间最短路径长度, $n-1$ 是节点 v 能够达到的节点个数.

将计算得到的3个节点中心性得分进行相加,经过 *Sigmoid* 非线性激活函数,得到节点在图拓扑结构维度下的重要性排名 $S_2 \in \mathbb{R}^{N \times 1}$:

$$S_2 = \sigma(C_D + C_B + C_C) \quad (9)$$

由于单个维度下的节点排名往往具有偏见,每个维度下的节点重要性可能不同,本文尝试将不同维度下的节点排名进行汇聚融合,以生成鲁棒性更好的节点排名.因此,使用注意力机制自动学习每个维度得分 $\{S_d\}_{d=1}^D$ 下的权重 $\{w_d\}_{d=1}^D$:

$$w_d = \frac{\exp(\sigma(hz_d + b_d))}{\sum_{k=1}^D \exp(\sigma(hz_k + b_k))} \quad (10)$$

其中, $w_d \in \mathbb{R}^{N \times 1}$ 为维度 d 下的权重矩阵, h 是不同维度下节点得分的拼接, z_d 和 b_d 是维度 d 下的参数向量, D 是维度的集合.因此权重 w_d 越大,维度 d 越重要.以权重为系数,得到最终的节点重要性得分 $S \in \mathbb{R}^{N \times 1}$:

$$S = \sigma \left(\sum_{d=1}^D w_d \cdot S_d \right) \quad (11)$$

在得到最终的节点重要性得分后,使用得分作为评价标准选择在池化过程中应该被保留下的节点.具体来说,首先根据节点重要性得分对节点进行排序,选择排名靠前的节点生成诱导子图.记录所选节点编号,用于生成下一层的邻接矩阵和特征矩阵,方法如下:

$$\begin{cases} idx = \text{rank}(S, [r \times N^l]) \\ \tilde{S} = S(idx, :) \\ \tilde{X}^l = X^l(idx, :) \\ A^{l+1} = A^l(idx, idx) \\ X^{l+1} = \tilde{X}^l \odot (\tilde{S} 1_d) \end{cases} \quad (12)$$

其中, r 是图中节点的池化率, $N^{l+1} = [r \times N^l]$ 代表新图中节点数目. $\text{rank}(\cdot)$ 表示返回排序后的前 N^{l+1} 个节点索引的函数. $S(idx, :)$ 表示根据索引保留诱导子图节点的得分, $A^l(idx, idx)$ 和 $X^l(idx, :)$ 依据索引进行矩阵行或列的提取,形成诱导子图的特征矩阵和邻接矩阵. $1_d \in \mathbb{R}^d$ 是一个长度为 d , 元素全部为1的向量, \odot 为哈达玛积,使矩阵逐元素相乘.因此,经过以上池化层操作,我们得到 $A^{l+1} \in \mathbb{R}^{N^{l+1} \times N^{l+1}}$, $X^{l+1} \in \mathbb{R}^{N^{l+1} \times d}$ 作为下一层输入的邻接矩阵和特征矩阵.

2.4 其他层

图神经网络层:图卷积网络在许多与提取图特征相关的机器学习任务中得到了最先进的性能^[8].因此我们使用GCN作为模型中提取特征信息的模块,其定义为:

$$X^{l+1} = \sigma(\hat{D}^{-\frac{1}{2}} \hat{A}^l \hat{D}^{\frac{1}{2}} X^l W^l) \quad (13)$$

其中, W^l 是第 l 层可学习的参数矩阵, X^l 是特征矩阵,初始节点在第1次进行卷积操作时使用 $X^0 = X_{\text{input}}$. $\hat{A}^l = A^l + I$ 是对每个节点添加自环后的邻接矩阵, \hat{D} 是添加了自环后的度矩阵.实际上我们提出的MDPool池化算子也可以与其他图神经网络模型结合,我们将在第3.4节实验部分讨论这些问题.

反池化层:模型中的编码部分使用池化层缩减图的规模,而在解码部分,图结构数据需要被上采样以恢

复其原本的节点规模. 为了实现图结构数据的上采样, 我们使用反池化层, 执行 MDPool 的反向操作. 如图 2 虚线所示, 在 MDPool 过程中保存图的邻接矩阵信息以及所选节点的索引. 如图 4 所示, 在反池化层中, 邻接矩阵直接使用对应编码层的邻接矩阵, 特征矩阵根据索引信息, 将现有特征放回图中原始位置, 其表述如下:

$$X^{l+1} = \text{unpool}(0_{N \times d}, X^l, \text{idx}) \quad (14)$$

其中, $X^l \in \mathbb{R}^{K \times d}$ 为待放回的节点特征, $\text{idx} \in \mathbb{R}^K$ 为相应池化层中的节点索引, $0_{N \times d}$ 是新图的初始化为 0 的特征矩阵. $\text{unpool}(\cdot)$ 函数按照 idx 中的索引将 X^l 的每一个行向量填充到 $0_{N \times d}$ 矩阵中去. 经过反池化操作我们将得到新图的特征矩阵 $X^{l+1} \in \mathbb{R}^{N \times d}$, 图中节点数目从 K 变成 N .

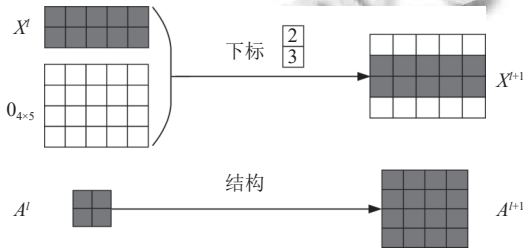


图 4 UnPool 层的说明

读出层: 如图 2 所示图分类框架由多次重复的卷积层池化层模块以及反池化层卷积层模块构成. 每个模块可以得到不同规模的子图特征: X^1, X^2, \dots, X^L , 由于不同模块输出特征的节点数目不同, 参考读出层的设计^[39], 将所有层节点特征汇聚在一起, 形成一个固定大小的表示:

$$\begin{cases} X_max = \left[\max_{i=1}^d X^l(:, i) \right]_{l=1}^{l=L} \\ X_mean = \left[\frac{1}{N^l} \sum_{j=1}^{N^l} X^l(j, :) \right]_{l=1}^{l=L} \end{cases} \quad (15)$$

其中, X_max 为框架中每层特征矩阵, 按特征列选取最大值形成的列表, X_mean 为每层特征矩阵中所有节点特征的均值形成的列表, 列表长度均为 L . 我们将不同规模的读出层输出进行拼接, 最终形成图形级别的表示:

$$Z = \sigma(X_max \parallel X_mean) \quad (16)$$

其中, $Z \in \mathbb{R}^{2Ld}$ 为图的特征, \parallel 为向量拼接操作.

损失函数: 我们将图的特征表示放入带有 $\text{Softmax}(\cdot)$

分类器的多层感知机中, 损失函数使用交叉熵损失:

$$\begin{cases} \hat{Y} = \text{Softmax}(MLP(Z)) \\ \text{loss} = - \sum_{i \in G_{\text{train}}} \sum_{j=1}^C Y_{ij} \log \hat{Y}_{ij} \end{cases} \quad (17)$$

其中, \hat{Y}_{ij} 表示训练集中图 i 预测为第 j 类的概率, Y_{ij} 为其真实标签, C 表示为类别数.

3 实验与分析

3.1 数据集及基线

本文采取了 4 个广泛使用的数据集来评估我们提出的图分类模型性能. 包含了生物信息学和社交网络领域的数据库^[40-43], 包括二分类任务和多分类任务, 4 个数据集的统计信息总结于表 1. DD 数据集^[40,41] 由蛋白质的结构图组成, 节点代表氨基酸, 边表示两个氨基酸之间距离小于 6 埃米, 标签为两类用于判断蛋白质是否为酶. PROTEINS 数据集^[41,42] 是蛋白质的图表示, 节点代表蛋白质的二级结构, 边表示蛋白质二级结构在氨基酸序列中为邻居或三维空间中是邻居. 标签为两类用于判断蛋白质是否为酶. COLLAB 是一个科学合作数据集^[43], 节点代表科研人员, 每一个图为科研人员的自我中心网络, 标签代表所属的研究领域. IMDB-MULTI 是一个电影合作数据集^[43], 节点代表演员, 每个图为演员的自我中心网络, 标签为电影类型.

表 1 实验数据集统计

| 数据集 | Graph数 | 类别数 | 平均节点数 | 平均边数 |
|------------|--------|-----|--------|---------|
| DD | 1178 | 2 | 284.32 | 715.66 |
| PROTEINS | 1113 | 2 | 39.06 | 72.82 |
| COLLAB | 5000 | 3 | 74.49 | 2457.78 |
| IMDB-MULTI | 1500 | 3 | 13.00 | 65.94 |

我们使用 3 类方法作为实验基线: (1) 基于图核的方法. 此类方法使用不同核方法来进行图分类, 包括 W-L 子树核 (1-WL)^[40], W-L 最优分配核 (WL-OA)^[44], Graphlet 核 (GR)^[45], 最短路径核 (SP)^[46]. (2) 基于图神经网络的方法, 包括 GCN^[8], ChebNet^[47], GraphSAGE^[9], GAT^[48]. (3) 基于池化的方法, 包括两个全局池化方法 Set2Set^[27] 和 DGCNN^[28], 以及 6 个层次图池化模型 DiffPool^[18], EdgePool^[21], SAGPool^[22], gPool^[20], ASAPool^[33], GSAPool^[34].

3.2 实验和参数设置

我们使用 PyTorch 以及 PyG 框架实现 EDMDPool 模型, 并使用 10 折交叉验证作为模型性能评估. EDMD-

Pool 中我们设置编码块解码块的个数均为 3, 根据数据集中节点规模, 将节点表示维度设置为 128 或 64. 根据数据集的规模, 将批次大小设置为 64 或 32. 设置多头自注意力头数为 2, 维度为 128. 使用具有两个全连接层的 MLP 后跟 *Softmax* 分类器做预测, 我们使用 Adam 优化器来优化模型参数, 学习率设置为 0.001, 对于不同数据集每层的池化率使用与 gPool 相同的设置, 选择 200 轮训练中最优的结果.

3.3 实验结果

我们在图分类任务的 4 个典型数据集上比较了提出的 EDMDPool 与基线方法的性能, 准确率以及标准差在表 2 给出, 粗体突出显示了最佳结果. 首先, 可以观察到, EDMDPool 的性能在 4 个数据集中始终优于其他经典基线的表现. 具体来说, 我们的模型在 DD、PROTEINS、IMDB-MULTI 数据集上分别比表现最好的模型提升了 1.52%、0.62%、1.8% 的准确率. 在 COLLAB 数据集上虽然 EDMDPool 与 SAGPool 模型准确率差别不明显, 但是 EDMDPool 取得了较低的标准差. 与其他模型相比, EDMDPool 在 COLLAB 数据集上的表现较为相对平庸. 并且可以看到, 我们提出的 MDPool_{Global}

全局池化的模型在 COLLAB 数据集上可以取得较好的表现. 那是因为 COLLAB 数据集中许多科研协作图只有单层的社团结构. EDMDPool 具有 3 组编码解码模块的复杂层次结构, 可能会导致过拟合现象, 这与之前工作发现的结果一致^[19].

值得注意的是, 传统的基于图核方法由于其非端到端的学习方式, 导致该类方法不能推广到任意结构的图上, 在部分数据集上效果较差. 此外, 与具有池化模块的框架相比, GNN 组均不能达到较好的效果, 主要原因是在汇聚全局节点特征时, 忽略了图的结构信息, 这进一步验证了在框架中增加池化层的必要性. 我们还可以看到, 大部分基于分层的池化方法效果超过了全局池化方法 Set2Set 和 DGCNN. 这是因为基于全局的池化方法, 本质上图的特征表示仍是“扁平的”, 图的层次结构信息未被考虑. 在大部分基线中, 层次池化模型可以获得更好的性能, 这说明了我们使用的层次池化方式的有效性. 对于全局池化方法, 我们同样设计了 MDPool_{Global} 作对比. 可以看到, 与基线模型相比较, 我们提出的全局池化方式仍能取得较好效果, 这进一步证明了我们提出池化层的有效性.

表 2 图分类数据集的平均准确率 (10 折) 和标准差 (%)

| 方法 | 模型 | DD | PROTEINS | COLLAB | IMDB-MULTI |
|------|--------------------------|-------------------|-------------------|-------------------|-------------------|
| 核方法 | WL | 78.35±3.93 | 72.74±4.61 | 78.14±1.79 | 49.98±4.26 |
| | WLOA | 78.33±3.51 | 73.44±4.20 | 79.90±1.66 | 49.82±3.84 |
| | GR | 78.31±4.26 | 71.60±4.35 | 64.43±2.02 | 39.77±3.60 |
| | SP | 78.58±3.97 | 75.70±4.04 | 59.20±2.82 | 39.17±4.59 |
| GNN | GCN | 76.56±3.91 | 72.88±3.77 | 78.46±1.66 | 52.8±4.98 |
| | ChebNet | 80.31±2.88 | 68.92±3.56 | 81.70±1.47 | 51.93±4.53 |
| | GraphSAGE | 81.07±1.61 | 70.18±3.53 | 78.84±1.73 | 52.93±4.35 |
| | GAT | 76.40±4.43 | 67.57±2.92 | 78.68±1.31 | 52.80±5.04 |
| 全局池化 | Set2Set | 73.68±2.85 | 70.00±5.22 | 81.08±2.14 | 53.27±4.96 |
| | DGCNN | 77.16±1.96 | 75.66±2.62 | 77.06±1.39 | 52.60±5.06 |
| 分层池化 | EdgePool | 78.44±3.57 | 77.45±3.38 | 79.52±2.06 | 53.07±4.75 |
| | DiffPool | 78.86±2.09 | 77.90±3.10 | 79.74±1.65 | 54.67±4.34 |
| | ASAPool | 80.90±2.78 | 77.01±3.26 | 80.70±1.30 | 54.27±5.00 |
| | gPool | 82.51±3.17 | 78.26±4.52 | 79.36±1.46 | 52.47±2.90 |
| | SAGPool | 79.88±1.65 | 76.47±5.16 | 82.48±1.90 | 54.40±5.03 |
| | GSAPool | 80.81±3.46 | 77.82±3.56 | 81.5±1.54 | 52.93±5.71 |
| 本文 | MDPool _{Global} | 82.35±3.6 | 77.10±4.86 | 83.72±1.03 | 55.20±3.23 |
| | EDMDPool | 84.13±2.52 | 78.88±4.35 | 82.49±0.98 | 56.47±3.80 |

3.4 消融实验及参数分析

为了验证各维度下的信息都有助于图分类, 我们将 EDMDPool 模型与其变体比较, 其中, 仅使用节点特征的模型为 EDFPool, 仅使用图拓扑结构的模型为 EDSPool, 全局池化模型为 MDGPool. 图 5 显示了各个

变体的分类效果, 在 DD、PROTEINS、IMDB-MULTI 这 3 个数据集上, 单一维度下的变体的性能均不如 EDMDPool, 这证明了多维度得分整合的有效性. 对于 COLLAB 数据集, 由于出现过拟合现象, EDMDPool 的准确率低于 EDSPool. 但是 MDGPool, 仍比仅使用单

一维度的EDSPool性能更好,这进一步证明了多维度信息融合进行节点选取的有效性。

为了证明本文提出的MDPool池化层可以集成到任意图神经网络框架中,我们选择了4种最广泛使用的图神经网络方法构建模型。在两个不同领域的数据集上对他们进行了评估,涵盖了二分类及多分类任务。结果如表3所示,根据所选择的数据集及GNN类型,图数据的分类性能有所不同,但不同GNN类型的模型都能得到相似的分类效果,这表明了我们提出的模型对各种图神经网络方法的兼容性较好。

我们进一步研究了一些重要的超参数对EDMDPool的影响,包括编码解码模块的深度 d ,学习率 r 。如图6所示,当分别设置 $d=3$ 和 $r=0.001$ 时,模型可以实现最佳性能。随着学习率的减小分类精度呈增长趋势,当学习率过小时,出现局部极值点收敛现象,精度开始降低。对于模块的深度,大的 d 可以提高分类结果,但过大则会导致过拟合,损害分类性能。

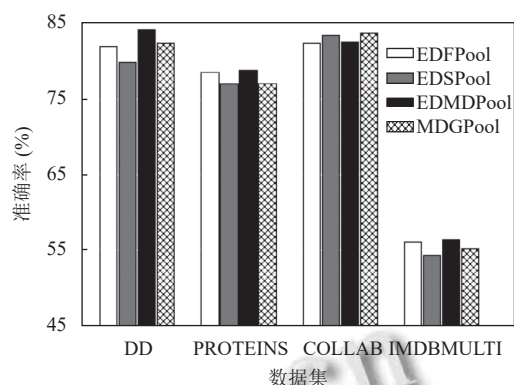


图5 多维度池化层不同变体下的性能

表3 MDPool在不同GNN框架下的表现(%)

| GNN结构 | PROTEINS | IMDB-MULTI |
|--------------------|------------|------------|
| EDMDPool-ChebNet | 78.35±4.22 | 55.8±2.72 |
| EDMDPool-GCN | 78.88±4.35 | 56.47±3.8 |
| EDMDPool-GraphSAGE | 78.62±4.09 | 55.33±3.00 |
| EDMDPool-GAT | 78.52±4.22 | 55.07±2.93 |

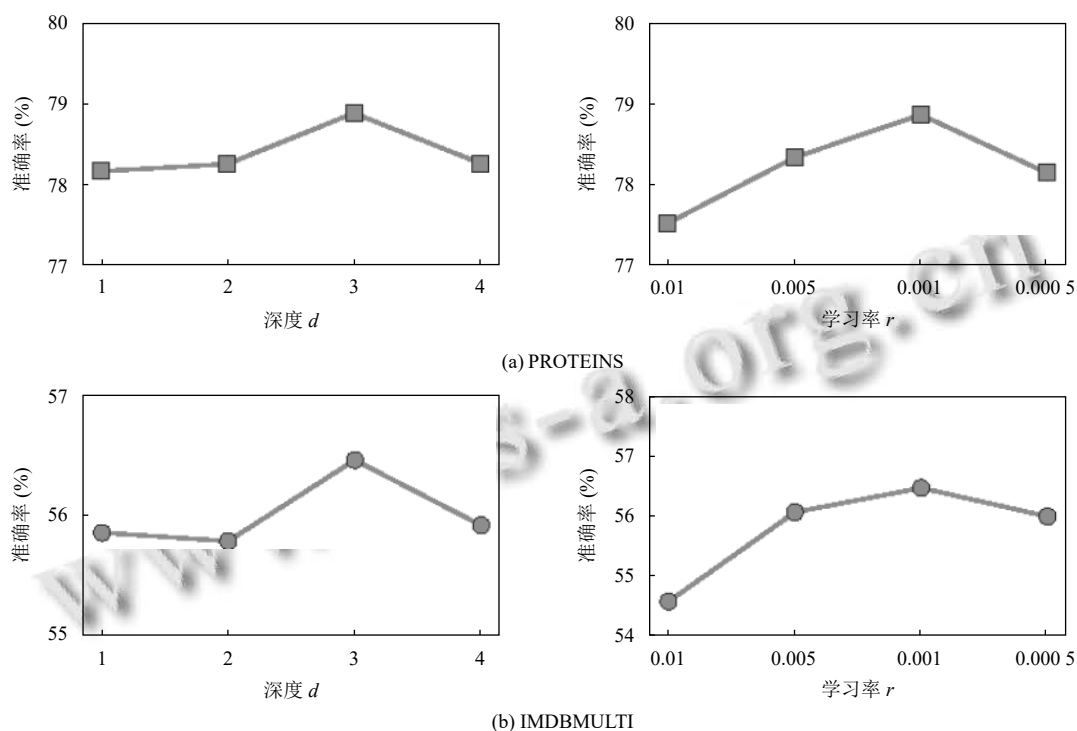


图6 不同深度和学习率在两数据集上的分类精度曲线

4 结论与展望

为了提高图分类任务的准确性,增强图分层表示的能力,提出了一种基于多维度信息节点选择的图池化算子MDPool。使用节点特征维度和图拓扑结构维度

下的信息分别评估节点重要性。由不同维度下的节点排名,协作生成鲁棒性更高的节点得分,依据排序后的得分选择节点构成诱导子图,实现对原始图的下采样操作。将MDPool池化层、UnPool反池化层与现有图

神经网络模型堆叠形成编码解码框架,构建了一个端到端的图分层表示学习模型 EDMDPool. 在4种数据集上进行了10倍交叉验证. 实验结果表明, EDMDPool 具有较好的性能.

参考文献

- 1 LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature*, 2015, 521(7553): 436–444. [doi: [10.1038/nature14539](https://doi.org/10.1038/nature14539)]
- 2 Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 2017, 60(6): 84–90. [doi: [10.1145/3065386](https://doi.org/10.1145/3065386)]
- 3 Yu SH, Wang M, Pang SC, *et al.* Intelligent fault diagnosis and visual interpretability of rotating machinery based on residual neural network. *Measurement*, 2022, 196: 111228. [doi: [10.1016/j.measurement.2022.111228](https://doi.org/10.1016/j.measurement.2022.111228)]
- 4 Davidson EH, Rast JP, Oliveri P, *et al.* A genomic regulatory network for development. *Science*, 2002, 295(5560): 1669–1678. [doi: [10.1126/science.1069883](https://doi.org/10.1126/science.1069883)]
- 5 Kazius J, McGuire R, Bursi R. Derivation and validation of toxicophores for mutagenicity prediction. *Journal of Medicinal Chemistry*, 2005, 48(1): 312–320. [doi: [10.1021/jm040835a](https://doi.org/10.1021/jm040835a)]
- 6 Lazer D, Pentland A, Adamic L, *et al.* Computational social science. *Science*, 2009, 323(5915): 721–723. [doi: [10.1126/science.1167742](https://doi.org/10.1126/science.1167742)]
- 7 Gori M, Monfardini G, Scarselli F. A new model for learning in graph domains. *Proceedings of the 2005 IEEE International Joint Conference on Neural Networks*. Montreal: IEEE, 2005. 729–734.
- 8 Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. *Proceedings of the 5th International Conference on Learning Representations*. Toulon: OpenReview.net, 2017.
- 9 Hamilton WL, Ying R, Leskovec J. Inductive representation learning on large graphs. *Proceedings of the 31st International Conference on Neural Information Processing Systems*. Long Beach: Curran Associates Inc., 2017. 1025–1035.
- 10 Schlichtkrull M, Kipf TN, Bloem P, *et al.* Modeling relational data with graph convolutional networks. *Proceedings of the 15th International Conference on Semantic Web*. Heraklion: Springer, 2018. 593–607.
- 11 Ying R, He RN, Chen KF, *et al.* Graph convolutional neural networks for web-scale recommender systems. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. London: ACM, 2018. 974–983.
- 12 王兆慧, 沈华伟, 曹婧, 等. 图分类研究综述. *软件学报*, 2022, 33(1): 171–192. [doi: [10.13328/j.cnki.jos.006323](https://doi.org/10.13328/j.cnki.jos.006323)]
- 13 Borgwardt KM, Kriegel HP, Vishwanathan SVN, *et al.* Graph kernels for disease outcome prediction from protein-protein interaction networks. *Proceedings of the Biocomputing 2007*. Maui: World Scientific, 2007. 4–15.
- 14 Fout A, Byrd J, Shariat B, *et al.* Protein interface prediction using graph convolutional networks. *Proceedings of the 31st International Conference on Neural Information Processing Systems*. Long Beach: Curran Associates Inc., 2017. 6533–6542.
- 15 Smalter A, Huan J, Lushington G. Graph wavelet alignment kernels for drug virtual screening. *Journal of Bioinformatics and Computational Biology*, 2009, 7(3): 473–497. [doi: [10.1142/S0219720009004187](https://doi.org/10.1142/S0219720009004187)]
- 16 Mahé P, Vert JP. Graph kernels based on tree patterns for molecules. *Machine Learning*, 2009, 75(1): 3–35. [doi: [10.1007/s10994-008-5086-2](https://doi.org/10.1007/s10994-008-5086-2)]
- 17 Wang XF, Li Z, Jiang MJ, *et al.* Molecule property prediction based on spatial graph embedding. *Journal of Chemical Information and Modeling*, 2019, 59(9): 3817–3828. [doi: [10.1021/acs.jcim.9b00410](https://doi.org/10.1021/acs.jcim.9b00410)]
- 18 Ying Z, You JX, Morris C, *et al.* Hierarchical graph representation learning with differentiable pooling. *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. Montreal: Curran Associates Inc., 2018. 4805–4815.
- 19 Du JL, Wang SZ, Miao H, *et al.* Multi-channel pooling graph neural networks. *Proceedings of the 30th International Joint Conference on Artificial Intelligence*. Montreal: IJCAI.org, 2021. 1442–1448.
- 20 Gao HY, Ji SW. Graph U-Nets. *Proceedings of the 36th International Conference on Machine Learning*. Long Beach: PMLR, 2019. 2083–2092.
- 21 Diehl F. Edge contraction pooling for graph neural networks. arXiv:1905.10990, 2019.
- 22 Lee J, Lee I, Kang J. Self-attention graph pooling. *Proceedings of the 36th International Conference on Machine Learning*. Long Beach: PMLR, 2019. 3734–3743.
- 23 Zhang Z, Bu JJ, Ester M, *et al.* Hierarchical graph pooling with structure learning. arXiv:1911.05954, 2019.
- 24 Chen HX, Yin HZ, Chen T, *et al.* Exploiting centrality information with graph convolutions for network representation learning. *Proceedings of the 2019 IEEE 35th*

- International Conference on Data Engineering (ICDE). Macao: IEEE, 2019. 590–601.
- 25 Zhang Z, Bu JJ, Ester M, *et al.* Hierarchical multi-view graph pooling with structure learning. *IEEE Transactions on Knowledge and Data Engineering*, 2023, 35(1): 545–559.
- 26 Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. *Proceedings of the 3rd International Conference on Learning Representations*. San Diego, 2015.
- 27 Vinyals O, Bengio S, Kudlur M. Order matters: Sequence to sequence for sets. *Proceedings of the 4th International Conference on Learning Representations*. San Juan, 2016.
- 28 Zhang MH, Cui ZC, Neumann M, *et al.* An end-to-end deep learning architecture for graph classification. *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*. New Orleans: AAAI, 2018. 4438–4445.
- 29 Diehl F, Brunner T, Le MT, *et al.* Towards graph pooling by edge contraction. *Proceedings of the 2019 ICML Workshop on Learning and Reasoning with Graph-structured Data*. 2019.
- 30 顾昕, 叶海良, 杨冰, 等. 结合信息保留的多头注意力图池化模型. *中国计量大学学报*, 2022, 33(2): 288–296. [doi: [10.3969/j.issn.2096-2835.2022.02.019](https://doi.org/10.3969/j.issn.2096-2835.2022.02.019)]
- 31 张红梅, 李浩然, 张向利. 基于重要性池化的层级图表示学习方法. *桂林电子科技大学学报*, 2020, 40(4): 300–304. [doi: [10.3969/j.issn.1673-808X.2020.04.007](https://doi.org/10.3969/j.issn.1673-808X.2020.04.007)]
- 32 朱小草, 郭春生, 张宏宽, 等. 基于稀疏注意力的自适应图池化方法. *杭州电子科技大学学报 (自然科学版)*, 2021, 41(5): 32–38.
- 33 Ranjan E, Sanyal S, Talukdar P. ASAP: Adaptive structure aware pooling for learning hierarchical graph representations. *Proceedings of the 34th AAAI Conference on Artificial Intelligence*. New York: AAAI, 2020. 5470–5477.
- 34 Zhang L, Wang XD, Li HS, *et al.* Structure-feature based graph self-adaptive pooling. *Proceedings of the 2020 Web Conference*. Taipei: ACM, 2020. 3098–3104.
- 35 Hu FY, Zhu YQ, Wu S, *et al.* Hierarchical graph convolutional networks for semi-supervised node classification. *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. Macao: AAAI, 2019. 4532–4539.
- 36 Vaswani A, Shazeer N, Parmar N, *et al.* Attention is all you need. *Proceedings of the 31st International Conference on Neural Information Processing Systems*. Long Beach: Curran Associates Inc., 2017. 6000–6010.
- 37 Brandes U. On variants of shortest-path betweenness centrality and their generic computation. *Social Networks*, 2008, 30(2): 136–145. [doi: [10.1016/j.socnet.2007.11.001](https://doi.org/10.1016/j.socnet.2007.11.001)]
- 38 Freeman LC. Centrality in social networks conceptual clarification. *Social Networks*, 1978, 1(3): 215–239. [doi: [10.1016/0378-8733\(78\)90021-7](https://doi.org/10.1016/0378-8733(78)90021-7)]
- 39 Xu K, Li CT, Tian YL, *et al.* Representation learning on graphs with jumping knowledge networks. *Proceedings of the 35th International Conference on Machine Learning*. Stockholm: PMLR, 2018. 5449–5458.
- 40 Shervashidze N, Schweitzer P, van Leeuwen EJ, *et al.* Weisfeiler-Lehman graph kernels. *The Journal of Machine Learning Research*, 2011, 12(9): 2539–2561.
- 41 Dobson PD, Doig AJ. Distinguishing enzyme structures from non-enzymes without alignments. *Journal of Molecular Biology*, 2003, 330(4): 771–783. [doi: [10.1016/S0022-2836\(03\)00628-4](https://doi.org/10.1016/S0022-2836(03)00628-4)]
- 42 Borgwardt KM, Ong CS, Schönauer S, *et al.* Protein function prediction via graph kernels. *Bioinformatics*, 2005, 21(S1): i47–i56.
- 43 Yanardag P, Vishwanathan SVN. Deep graph kernels. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Sydney: ACM, 2015. 1365–1374.
- 44 Kriege NM, Giscard PL, Wilson RC. On valid optimal assignment kernels and applications to graph classification. *Proceedings of the 30th International Conference on Neural Information Processing Systems*. Barcelona: Curran Associates Inc., 2016. 1623–1631.
- 45 Shervashidze N, Vishwanathan SVN, Petri T, *et al.* Efficient graphlet kernels for large graph comparison. *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics*. Clearwater Beach: JMLR.org, 2009. 488–495.
- 46 Borgwardt KM, Kriegel HP. Shortest-path kernels on graphs. *Proceedings of the 5th IEEE International Conference on Data Mining*. Houston: IEEE, 2005. 74–81.
- 47 Defferrard M, Bresson X, Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering. *Proceedings of the 30th International Conference on Neural Information Processing Systems*. Barcelona: Curran Associates Inc., 2016. 3844–3852.
- 48 Veličković P, Cucurull G, Casanova A, *et al.* Graph attention networks. *Proceedings of the 6th International Conference on Learning Representations*. Vancouver: OpenReview.net, 2018.

(校对责编: 孙君艳)