

图神经网络进阶

引言

图神经网络作为利用深度学习处理图结构数据的前沿工具,在捕捉节点关系、结构信息和非欧几里得特性方面展现了强大的能力。然而,由于图数据和应用场景的复杂化,传统图神经网络模型在性能和适用性上暴露出了一些瓶颈,往往受到数据质量、模型架构和训练策略的限制。随着图神经网络的发展,越来越多的先进模型被提出以突破传统图神经网络的瓶颈。本章从数据、架构和训练三个层面介绍了针对传统图神经网络的优化策略。在数据层面,介绍如何提升数据质量,挖掘图数据潜在的高质量特征,减少噪声干扰;在架构层面,介绍如何改进传统图神经网络主要部件,包括消息传递,采样和池化,以更好地捕捉图数据的复杂关系,提升模型性能;在训练层面,介绍如何改进图神经网络的训练策略,包括图自监督学习和图课程学习,从而缓解依赖大量标注进行训练的困境,并且实现更稳健的模型收敛和更高的泛化能力。

本章学习目标

- (1) 理解不同图数据优化技术的目的,掌握从结构、特征和标签三个方面对图数据进行优化的主要方法;
- (2) 理解传统图神经网络在消息传递方面存在的主要问题,掌握针对过平滑、长距离依赖,表达能力受限问题的主要解决方法;
- (3) 掌握图上的采样和池化的主要改进方法,并能根据实际任务灵活选择;
- (4) 掌握在缺乏标签情况下设计对比或生成式图自监督训练的方法;
- (5) 了解图课程学习的主要思想,学会利用图课程学习来稳定训练和提升模型效果。

5.1 数据优化

在机器学习的发展历程中,丰富的实践经验表明,大量高质量数据是提升模型性能、推动模型进步的关键因素。例如,大规模视觉数据集 ImageNet 的引入,催生出了经典的卷积神经网络模型 AlexNet 和 ResNet,奠定了卷积神经网络在

图像处理领域的主导地位。同样,在图学习领域,大量研究工作聚焦于如何改进图数据质量,以便图神经网络能够更好地捕捉图数据中的信息。这些研究主要从图结构优化、图特征优化和图标签优化三个方面入手,在提升图神经网络模型的准确性、鲁棒性和效率方面发挥了重要作用。

5.1.1 图结构优化

图结构是图数据中最核心的部分,它描述了节点之间的关联信息。本节将探讨如何从图结构的角度优化图数据。首先介绍结构缩减,旨在减少图中的冗余节点和边,以降低计算复杂度并提高模型的可扩展性;其次介绍结构增强,通过较低的开销丰富图结构信息,从而缓解模型过拟合的问题;再次介绍结构生成,其目标是生成高质量且多样化的图样本;最后介绍结构学习,专注于从图数据中挖掘出有价值的图结构,进一步提升图模型的表达能力。

1. 结构缩减

近年来,图数据集的规模和复杂性呈现出指数级增长。对于大规模网络(如社交图和引文网络),现有图神经网络在可扩展性和效率方面面临着严峻挑战。结构缩减技术在保留关键信息的前提下,通过减少图数据集的规模,降低计算复杂度并提升模型可扩展性。结构缩减方法可以分为三类。

1) 图稀疏化

图稀疏化通过移除原始图 G 中的部分边,生成一个简化图 $G_s = \{V, E_s\}$,其中 $E_s \subseteq E$ 。通常 G_s 需要保持原图 G 的某些关键性质,如图切割值、最短路径等。割稀疏化(Cut Sparsification)是图稀疏化的一个典型例子。它的目标是在尽可能保持图切割值的前提下,通过减少图中的边来简化图结构。图的切割将图的节点分为两部分,可以表示为 $C = (V', V - V')$ 。在此过程中,图切割值(记作 $w(C)$)是指跨越该切割的所有边的权值和,这个值反映了两部分节点之间的连接强度。通常, ϵ -割稀疏化(ϵ -cut Sparsifier)是指一个简化图 G_s ,其图切割值在所有切割的情况下与原始图 G 的图切割值保持接近,表示如下:

$$(1-\epsilon)w_G(C) \leq w_{G_s}(C) \leq (1+\epsilon)w_G(C)$$

其中, ϵ 的取值范围为 $(0, 1)$,当 $\epsilon=0$ 时, G_s 完全等同于原始图 G ,随着 ϵ 增大,允许的偏差更多,但简化图 G_s 在图切割值上依然与原始图 G 保持接近。割稀疏化也广泛应用于解决图的连通性问题、最大流问题、最小二等分问题等。

2) 图粗化

图粗化通过将一组紧密连接的节点合并成超级节点的方法来简化图结构。简化图 G_s 可以记为 $G_s = (V_s, E_s)$,其中节点数 $|V_s| < |V|$ 。近年来,谱保持的图粗化方法(Spectrum-preserving Coarsening)因为可以较好地保持原始图的重要结构信息受到了较多关注。受限谱相似性(Restricted Spectral Similarity, RSS)^[1]是一种用来确保简化图能够学习到原始图谱特性的技术,具体可以定义为

$$(1-\zeta_k)\lambda_k \leq \mathbf{u}_k^T \tilde{\mathbf{L}} \mathbf{u}_k \leq (1+\zeta_k)\lambda_k$$

其中, λ_k 和 \mathbf{u}_k 分别表示原始图 G 的拉普拉斯矩阵 \mathbf{L} 的第 k 个特征值和特征向量, $\tilde{\mathbf{L}} \in \mathbf{R}^{N \times N}$ 是简化图 G_s 拉普拉斯矩阵的近似, ζ_k 是误差容忍度,通常取值范围为 $(0, 1)$ 。受限谱相似性方法通过确保简化图的拉普拉斯矩阵特征值和特征向量在一定误差范围内接近原始图,

从而确保简化图能有效地学习和保留原始图的谱特性。

3) 图压缩

不同于图稀疏化和图粗化在原始图上进行结构上的缩减,图压缩通过合成一个新的更小的图来实现原始图的压缩。图压缩的目的是生成一个包含较少节点和边的简化图 $G_s = \{V_s, E_s\}$, 其中 $|V_s| \ll |V|$, 使在简化图上训练的模型能表现出与原始图上训练的模型相似的性能。经典的图压缩框架是利用梯度匹配的方法对齐原始图和简化图的梯度^[2]。具体来说,它通过最小化梯度之间的距离,使在简化图上训练模型时,模型的参数更新与在原始图上训练时相似,可以表示如下:

$$\begin{aligned} \mathbf{r}_s &= \nabla_{\theta} L(f_{\theta}(\mathbf{A}_s, \mathbf{X}_s), \mathbf{Y}_s) \\ \mathbf{r} &= \nabla_{\theta} L(f_{\theta}(\mathbf{A}, \mathbf{X}), \mathbf{Y}) \\ \min \text{Dis}(\mathbf{r}_s, \mathbf{r}) \end{aligned}$$

其中, f_{θ} 表示图模型, θ 是模型参数,模型的输入是图的邻接矩阵 \mathbf{A} 和节点特征矩阵 \mathbf{X} , L 是损失函数, \mathbf{r}_s 是在简化图 G_s 上计算得到的梯度, \mathbf{r} 是在原始图 G 上计算得到的梯度, $\text{Dis}(\mathbf{r}_s, \mathbf{r})$ 是衡量两个梯度距离的函数,如余弦相似度。通过最小化此距离,简化图保留了原始图的结构和特征信息。

2. 结构增强

模型通常需要大量数据才能有效地理解数据的特征和规律。然而,由于图数据的稀缺性和稀疏性,图神经网络在训练过程中往往无法充分拟合图数据的底层分布,容易陷入局部最优解,导致模型过拟合,严重削弱了模型在实际应用中的有效性和可靠性。为了缓解这一问题,结构增强方法在不改变图关键信息的前提下对图的拓扑结构进行适当的扰动,以一种低开销的方式增强了拓扑结构信息,有效提高了模型的泛化能力。结构增强可分为启发式增强和自适应增强方法。

启发式增强方法是一类通过预定义规则或经验策略对图结构进行修改或扩展的技术,其核心思想是利用简单、高效的提升模型的鲁棒性和泛化能力。这些方法通常与具体任务无关,具有较强的普适性,同时对模型结构无显著依赖,因此容易集成到现有的图神经网络中。

1) 启发式增强

(1) 丢弃。

丢弃是结构增强中一种基本而广泛应用的技术,旨在通过随机丢弃图中的边、节点和特征等来改善模型的训练效果。这种方法通常不需要对模型结构进行修改,而是在训练过程中动态地对图进行随机裁剪,因此非常易于集成到现有的图神经网络训练流程中。以随机丢弃边的方法为例,DropEdge^[3]在每个训练轮次,在图的边集合中随机选择 $p|E|$ 条边进行丢弃,其中 p 表示去边率,经过丢弃后的邻接矩阵可以表示如下:

$$\mathbf{A}_{\text{drop}} = \mathbf{A} - \mathbf{A}'$$

其中, \mathbf{A}' 是随机选取的 $p|E|$ 条边组成的稀疏邻接矩阵。DropEdge 没有改变邻居聚合的期望,是一种无偏的图结构增强技术,类似于典型的图像增强操作(如旋转和裁剪),所以可以缓解过拟合问题,增强模型的泛化能力。

(2) 子图替换。

子图替换是为了弥补基于丢弃的方法仅关注到节点或边等最基本层次的信息,而忽视

了更高层次的信息的不足而提出的,它通过替换图中的特定子结构来实现图结构的增强。MoCL^[4]是在生物医学领域运用子图替换的方法进行图增强的一个重要的工作,它指出大多数丢弃的方法在增强过程中可能会改变分子图的语义,因此通过注入领域知识来辅助增强过程。MoCL引入了生物电子等排体(Bioisosteres)的概念,来对分子中的特定有效子结构进行替换。生物电子等排体是一类具有相似物理或化学性质的分子片段,替换后不会显著改变分子的整体性质。这种替换能够保持分子的生物活性和物理化学性质,同时引入变化以增强数据多样性。

2) 自适应增强

启发式增强方法还包括图扩散等,这里不做详细介绍。对于在特定任务上需要增强模型鲁棒性和性能时,启发式增强方法可能存在不足。自适应增强方法在训练阶段基于具体任务自适应地进行结构增强,分为基于边的方法、基于子图的方法,以及自动化增强方法。

(1) 基于边的方法。

为了使可微损失函数指导边增强过程,一些研究将图上边的权值视为可被优化的连续变量,而不是具有固定的边连接。这些工作通过引入特定的约束(如平滑性和稀疏性)来构建损失函数,从而生成用于优化边权重的梯度。例如,Pro-GNN^[5]提出了如下的损失函数:

$$L = \|\tilde{\mathbf{A}} - \mathbf{A}\|_F^2 + \eta \|\tilde{\mathbf{A}}\|_1 + \beta \|\tilde{\mathbf{A}}\|_* + \rho(\mathbf{X}^T \hat{\mathbf{L}} \mathbf{X}) + \gamma L_{\text{GNN}}$$

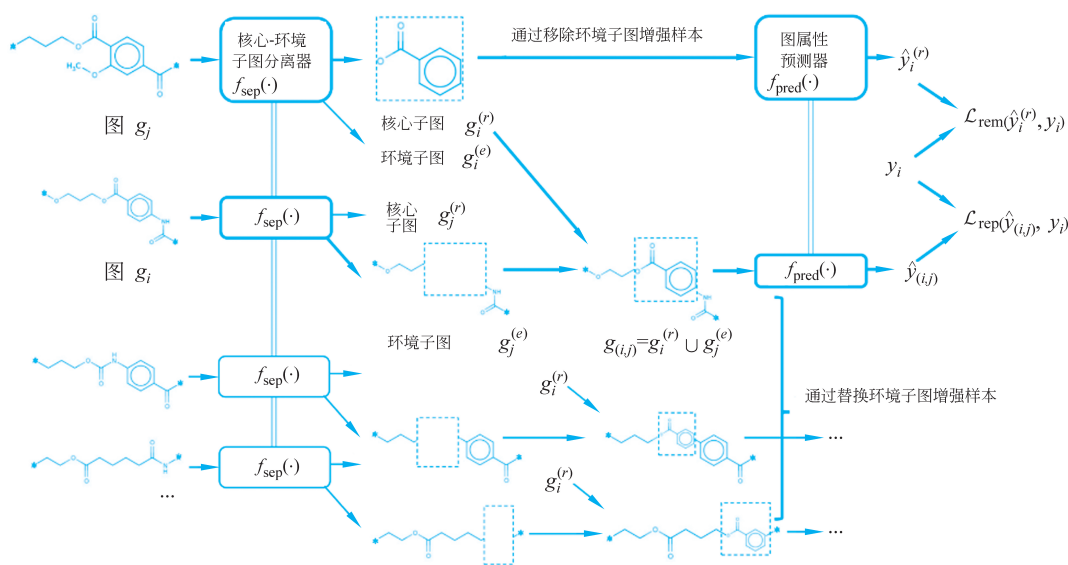
其中, $\tilde{\mathbf{A}}$ 是增强后的邻接矩阵, $\hat{\mathbf{L}}$ 是归一化的拉普拉斯矩阵。具体来说, $\|\tilde{\mathbf{A}} - \mathbf{A}\|_F^2$ ($\|\cdot\|_F$ 代表 Frobenius 范数)旨在让 $\tilde{\mathbf{A}}$ 接近原始邻接矩阵 \mathbf{A} 。 $\eta \|\tilde{\mathbf{A}}\|_1$ ($\|\cdot\|_1$ 表示 L_1 范数)和 $\|\tilde{\mathbf{A}}\|_*$ ($\|\cdot\|_*$ 表示核范数)分别确保图的稀疏性和低秩特性。此外, $\rho(\mathbf{X}^T \hat{\mathbf{L}} \mathbf{X})$ 控制特征的平滑性。 γ 控制针对具体任务的图神经网络损失函数 L_{GNN} 的比重。

(2) 基于子图的方法。

基于子图的自适应增强方法旨在找到最具代表性和信息量的子图,类似于分子中的官能团。然后基于这些子图来进行数据增强,以提高模型的性能、可解释性和鲁棒性等。GRE^[6]是一个典型的基于子图的增强过程,它定义了核心子图(Rationale Subgraph)和环境子图(Environment Subgraph)的概念,利用核心子图与不同的环境子图的组合生成新的数据样本,以让模型感知到核心子图的重要性,帮助模型学习更丰富的特征和拓扑结构。具体而言,如图5-1所示,核心子图(Rationale Subgraph)是指图结构中最能解释或支持模型预测的子结构。环境子图(Environment Subgraph)是指在核心子图被识别并分离后,图中剩余的部分。GRE^[6]首先利用图神经网络生成潜在节点表示,然后通过多层感知机计算掩码向量,通过优化属性预测损失来指示哪些节点属于核心子图。接着,它将核心子图与其他样本的环境子图的表示结合,以产生新的增强数据。通过这种方式,图的全局结构得到了丰富,同时保留了与任务相关的关键信息。在模型训练过程中,GRE^[6]的增强数据可以被视为新的输入,并参与优化模型参数。

(3) 自动化增强。

不同的数据集可能需要不同的增强策略,同一数据集的不同训练阶段也可能需要不同的增强策略。与上面两种固定的增强策略不同,自动化增强是一种在模型训练过程中动态学习最佳增强策略的方法,可以根据数据的特性和任务的需求调整其策略,以提高模型自适应数据分布和任务的能力。这种方法的主要思路是通过双层优化算法或强化学习来选择最

图 5-1 GREA 整体框架图^[6]

合适的增强方式。JOAO^[7]方法通过双层优化的框架来同时学习模型的编码器和增强策略,以实现更有效的数据增强。在内层优化过程中,首先使用当前的增强策略对训练数据进行增强,生成新的训练样本。这些增强的数据被用于训练模型的编码器,以提高其性能。在外层优化中,评估编码器性能并根据评估结果调整增强策略。JOAO 从具有可学习参数的分布中采样增强策略(如基于边和基于子图的方法),通过优化过程自动更新这些参数,以充分发挥每个增强策略的性能。

3. 结构生成

尽管图增强可以初步丰富拓扑信息,但不可避免地会引入噪声,从而影响模型性能。作为一种更高级的方法,图生成旨在生成高质量和多样化的图样本。在这一部分,将针对不同的生成的图数据形式介绍相应的代表性生成方法,包括节点序列的生成、邻接矩阵的生成和节点嵌入的生成。

1) 节点序列生成

将图简化为序列是图生成的一个初步思路,催生了自回归图生成方法。通常,自回归方法旨在基于预先采样的节点顺序逐个生成图的节点。然而,由于图的非唯一性和高维特性,使用节点顺序作为输入时需要考虑置换不变性的问题。为了解决这一挑战,GraphRNN^[8]提出使用广度优先搜索(Breadth-first Search, BFS)或深度优先搜索(Deep-first Search, DFS)来保证节点顺序的一致性,它的核心思想是将图的生成过程分为节点生成和边生成两个步骤。具体来说,GraphRNN 首先通过 BFS 或 DFS 遍历图,得到节点的序列。这一步使模型可以根据节点的访问顺序逐步生成图结构,确保生成过程具有唯一性。GraphRNN 在节点生成阶段逐个生成图中的节点。生成的节点数量可以根据图的规模动态调整。每个节点通过一个循环神经网络生成,该 RNN 根据当前生成的节点序列预测下一个节点。在生成了一个新节点之后,边生成器会预测该节点与已经生成的节点之间的连接情况。GraphRNN 使用另一个 RNN 模型对边的存在性进行逐一预测,判断是否存在边连接到当前已生成的节点。通过这种方式,GraphRNN 将图的生成转化为一个序列化的任务,避免

了直接处理图中所有节点和边带来的高计算成本。

2) 邻接矩阵生成

除了序列生成,另一种自然的思路是直接生成图的邻接矩阵。与序列生成的方法不同,这种方法一次性生成整个图的结构信息,尤其适用于小型图的生成。属于这一类别的方法,如 EDGE^[9],它的核心思想是基于离散扩散模型(Discrete Diffusion Model)。扩散模型最早用于连续空间数据的生成(如图像生成),而 EDGE 方法将其扩展到离散空间,应用于图的生成。它的工作原理主要分为两个阶段:第一阶段是正向扩散过程,逐步向图结构中加入噪声,直到数据接近于均匀噪声;第二阶段是反向去噪过程,从噪声数据开始,通过逐步去噪恢复出图结构信息。另外,EDGE 还为每个节点设计一个目标度向量,通过优化度匹配损失函数来确保生成的图符合预定的节点度分布。

3) 节点嵌入生成

生成图的邻接矩阵通常耗时较长,且无法扩展 to 大型图。一个可能的解决方案是间接生成图。例如,可以通过节点嵌入来表示邻接矩阵: $\mathbf{A} = \mathbf{H} \cdot \mathbf{H}^T$ 。通过这种方式,只需生成一个较小的张量 $\mathbf{H} \in \mathbf{R}^{N \times d}$,而不是大型邻接矩阵 $\mathbf{A} \in \mathbf{R}^{N \times N}$,其中 $d \ll N$ 。基于这一思想,变分图自编码器(Variational Graph Autoencoder, VGAE)^[10]首先利用编码器(通常是图神经网络)学习节点表示,之后与标准的变分自编码器(Variational Autoencoder, VGA)类似, VGAE 假设节点的潜在表示为服从高斯分布的随机变量,并通过参数矩阵的变换输出其均值和方差,然后利用重参数技巧重新采样潜在表示。解码器从潜在表示中重构图的邻接矩阵,这一步通常通过计算节点潜在表示的内积来预测边的存在概率。在训练过程中,通过优化链接预测损失函数重建图结构。

4. 结构学习

现实世界中的图结构往往噪声较多或不完整,导致模型的学习效果下降,甚至产生错误的结果。这种图结构在社交网络、交通系统、生物网络等多个领域中普遍存在。图结构学习旨在从图数据中发现和优化有价值的结构,以增强图表示学习。根据是否考虑边的权重信息,现有的图结构学习方法大致可以分为两类:离散图结构学习和加权图结构学习。

1) 离散图结构学习

离散图结构学习将图结构视为随机变量,进而可以从概率邻接矩阵中进行采样。这种方法的核心在于利用概率模型捕捉图中节点之间的关系,并通过采样得到多样化的图结构,以建模节点连接关系的潜在不确定性和多样性。在这一框架下,研究人员使用多种技术来联合优化概率邻接矩阵和图神经网络的参数。通过同时优化这两个部分,模型不仅能够更好地学习图中的复杂关系,还能提升其泛化能力。接下来,将介绍一种经典的基于蒙特卡洛方法的离散图结构学习方法。蒙特卡洛方法是一类基于随机抽样的计算方法,常用于数值积分、概率分布的近似、优化等问题。它的核心思想是通过大量随机样本来估计一个期望值或者积分结果,这种方法特别适合高维复杂问题,尤其是当解析解难以获得时。

在离散图结构学习中, VGCN^[11]通过参数化的随机图模型纳入不确定的图信息,并利用蒙特卡洛方法进行近似。具体而言, VGCN 的目标是通过已知的信息(包括部分已知标签、特征和观察到的图结构)推断节点或图的标签的后验概率,公式如下:

$$p(\mathbf{Z} | \mathbf{Y}_L, \mathbf{X}, G_{\text{obs}}) = \int p(\mathbf{Z} | \mathbf{W}, G, \mathbf{X}) p(\mathbf{W} | \mathbf{Y}_L, \mathbf{X}, G) p(G | \lambda) p(\lambda | G_{\text{obs}}) d\mathbf{W} dG d\lambda$$

其中, $p(\mathbf{Z}|\mathbf{W}, G, \mathbf{X})$ 表示在给定神经网络权重 \mathbf{W} 、图 G 和特征 \mathbf{X} 的情况下, 模型输出 \mathbf{Z} 的条件概率。这个概率可以通过图卷积神经网络来建模; $p(\mathbf{W}|\mathbf{Y}_L, \mathbf{X}, G)$ 表示在已知部分标签 \mathbf{Y}_L 、特征 \mathbf{X} 和图结构 G 的情况下, 图神经网络权重 \mathbf{W} 的后验概率, 这描述了在图神经网络上进行训练时, 权重的更新过程; $p(G|\lambda)$ 表示在给定参数 λ 的情况下, 随机图 G 的生成概率; $p(\lambda|G_{\text{obs}})$ 表示在给定观察到的图 G_{obs} 的情况下, 参数 λ 的后验概率, 它表示从观察到的图中推断出随机图生成模型的参数。上述积分通常是不可解析的, 因此通常需要采用近似方法来进行计算。VGCN 使用蒙特卡洛方法近似上述积分:

$$p(\mathbf{Z}|\mathbf{Y}_L, \mathbf{X}, G_{\text{obs}}) \approx \frac{1}{V} \sum_v \frac{V}{N_G S} \sum_{i=1}^{N_G} \sum_{s=1}^S p(\mathbf{Z}|\mathbf{W}_{s,i,v}, G_{i,v}, \mathbf{X})$$

蒙特卡洛近似通过采样的方法来估计标签 \mathbf{Z} 的后验分布 $p(\mathbf{Z}|\mathbf{Y}_L, \mathbf{X}, G_{\text{obs}})$ 。首先, 从图生成模型的参数分布 $p(\lambda|G_{\text{obs}})$ 中生成 V 个样本 λ_v , 这些样本代表不同的模型参数; 其次, 对于每个 λ_v , 从条件分布 $p(G|\lambda_v)$ 中生成 N_G 个图样本 $G_{i,v}$, 反映出在这些参数下可能的图结构; 再次, 对于每个图样本 $G_{i,v}$, 从贝叶斯图卷积神经网络中采样权重矩阵 $\mathbf{W}_{s,i,v}$; 最后, 通过对所有样本的加权平均, 近似计算出 $p(\mathbf{Z}|\mathbf{Y}_L, \mathbf{X}, G_{\text{obs}})$, 实现了综合考虑多种可能的图结构和神经网络权重来估计标签的后验概率。

2) 加权图结构学习

离散图结构学习过于依赖已知的图结构和节点连接模式, 缺乏对新节点的适应能力, 因此在处理未知或未见节点时表现不佳, 在推理阶段面对未见节点时, 往往无法有效进行归纳学习。另外, 与二元邻接矩阵相比, 加权邻接矩阵能够编码更丰富的边的信息, 有利于后续图表示学习。加权图结构学习往往假设节点属性或多或少包含推断图的隐性拓扑结构的有用信息, 因此将图结构学习作为定义在节点嵌入空间上的相似性度量学习, 使学到的相似性度量函数以后可以应用于未见过的节点嵌入集来推断图结构, 从而实现归纳图结构学习。

加权图结构学习的核心思想是基于节点嵌入学习节点对的相似性度量函数, 得到边的权值, 从而实现加权的图结构学习。最简单的度量函数是计算任意一对节点嵌入之间的点积, 可以表示为: $\mathbf{S}_{i,j} = \tilde{\mathbf{v}}_i^T \tilde{\mathbf{v}}_j$ 其中, $\mathbf{S} \in \mathbf{R}^{n \times n}$ 是一个节点相似性矩阵, $\tilde{\mathbf{v}}_i$ 和 $\tilde{\mathbf{v}}_j$ 是节点的向量表示。为了提高点积的学习能力, 引入具有可学习参数的点积: $\mathbf{S}_{i,j} = (\tilde{\mathbf{v}}_i \odot \tilde{\mathbf{u}})^T \tilde{\mathbf{v}}_j$, 其中 \odot 表示逐元素相乘, $\tilde{\mathbf{u}}$ 是一个非负的可训练权重向量, 用于强调节点嵌入的不同维度。需要注意的是, 输出的相似性矩阵 \mathbf{S} 是不对称的。为了进一步提高表达能力, 出现了引入权重矩阵的度量方法, $\mathbf{S}_{i,j} = \text{ReLU}(\mathbf{W}\tilde{\mathbf{v}}_i)^T \text{ReLU}(\mathbf{W}\tilde{\mathbf{v}}_j)$, 其中 \mathbf{W} 是一个 $d \times d$ 的权重矩阵, $\text{ReLU}(\mathbf{x}) = \max(0, \mathbf{x})$ 是一种激活函数, 在这里用于保证相似性矩阵的稀疏性。之后的方法进一步对两个节点嵌入应用了不同的线性变换, 并引入了归一化操作: $\mathbf{S}_{i,j} = \text{softmax}((\mathbf{W}_1 \tilde{\mathbf{v}}_i)^T \mathbf{W}_2 \tilde{\mathbf{v}}_j)$, 其中 \mathbf{W}_1 和 \mathbf{W}_2 是 $d \times d$ 的权重矩阵, softmax 函数定义为 $\text{softmax}(\vec{\mathbf{z}})_i = \frac{e^{z_i}}{\sum_j e^{z_j}}$, 用于获得行归一化的相似性矩阵。

5.1.2 图特征优化

图特征是用于描述图中节点、边或整个图的属性和信息的特征表示。在本节中, 将探讨如何从图特征的角度优化图数据。首先, 介绍特征增强, 通过扩展或修改原始特征, 避免模

型训练时的过拟合;其次,讨论特征选择,旨在识别和提取与标签高度相关的特征,避免维度灾难;最后,介绍特征补全,解决图数据中特征不完整的问题。

1. 特征增强

特征增强通过对节点、边或图的原始特征进行扩展或修改,为图数据引入额外的、相关的信息,不仅能够缓解模型的过拟合,还可以提高模型的泛化性能,特别是在处理复杂图结构或稀疏图数据时表现尤为显著。本节首先介绍一些通用的特征增强方法,然后介绍一种提高图神经网络表达能力的重要特征增强方法,即位置编码。

1) 通用特征增强

通用特征增强一般用于特征预处理,用于提高特征的效用和多样性,使特征更好地捕捉图信息。对于节点本身具有特征的情况,特征损坏(Feature Corruption)通过向原始节点特征中加入可控噪声来产生增广数据,可以表示为 $\tilde{\mathbf{X}} = \mathbf{X} + \mathbf{R}$,其中 \mathbf{X} 表示原始节点特征矩阵, \mathbf{R} 表示添加的噪声矩阵;特征重排(Feature Shuffling)通过随机切换特征矩阵中的行和列来改变原始节点特征的上下文信息,产生增广数据,形式化表示为 $\tilde{\mathbf{X}} = \mathbf{P}_r \mathbf{X} \mathbf{P}_c$,其中, \mathbf{P}_r 和 \mathbf{P}_c 分别是行排列矩阵和列排列矩阵,它们在每一行和每一列中恰好有一个元素为 1,其他位置均为 0;特征掩码(Feature Masking)的核心操作是将节点特征矩阵中的一部分置零,通过与一个 0-1 掩码矩阵 \mathbf{M} 逐元素相乘来实现: $\tilde{\mathbf{X}} = \mathbf{X} \odot \mathbf{M}$;特征添加(Feature Addition)通过将节点特征中缺少的节点属性编码成特征向量并与原始节点特征拼接实现,一般来说,可以表示为 $\tilde{\mathbf{x}}_i = [\mathbf{x}_i \parallel \mathbf{x}_j]$,其中 \parallel 表示拼接操作, \mathbf{x}_j 可以是一个空向量;特征传播(Feature Propagation)通过在图中扩散特征,结合来自不同节点的特征信息,表达式为 $\tilde{\mathbf{X}} = \tilde{\mathbf{A}} \mathbf{X}$,其中 $\tilde{\mathbf{A}}$ 是不同图传播方法对应的邻接矩阵。对于节点本身没有特征的情况,往往通过将图自身的结构信息编码成节点特征,例如直接使用节点的度作为节点特征。更加复杂的方法包括利用随机游走算法来捕获结构信息,并仿照 word2vec 技术来生成节点特征等。

2) 位置编码

众所周知,图神经网络的表达能力受到 1-WL 测试的限制,其在区分图同构性方面存在一定局限性,仅能够区分部分同构图。为了打破这一限制,一种常见的策略是通过引入位置信息来增强节点特征,即位置编码。这里将介绍两类位置编码方法:绝对位置编码和相对位置编码。

绝对位置编码(Absolute Position Encoding, APE)的目标是为每个节点分配一个位置表示,以指示其在整个图中的唯一位置。一种流行的 APE 方法是利用图拉普拉斯矩阵的特征向量,具体来说,通过对拉普拉斯矩阵 $\mathbf{\Delta}$ 进行特征分解,可以得到

$$\mathbf{\Delta} = \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2} = \mathbf{U}^T \mathbf{\Lambda} \mathbf{U}$$

其中, \mathbf{U} 是特征向量矩阵, $\mathbf{\Lambda}$ 是特征值矩阵,一般来说,选择前 k 个最小的非零特征值对应的特征向量,形成一个 $N \times k$ 的矩阵,每一行可看作一个节点的位置编码。通过将节点的位置编码与节点的原始特征组合,实现对原始特征的增强。

另外一种位置编码方式是相对位置编码(Relative Position Encoding, RPE),它通过将节点之间的距离作为位置编码来捕捉它们之间的关系信息。首先随机选择一个节点作为锚点节点,然后通过目标节点与锚点节点之间的相对距离来定位目标节点。如图 5-2(a)所

示,使用图神经网络往往无法区分节点 v_1 和 v_2 ,因为两个节点虽然位置不同,但是所处位置图结构相同。而通过选择 s_1 作为锚点节点,便可以通过计算和 s_1 的相对距离(跳数)来区分 v_1 和 v_2 。为了更精确捕捉节点的位置信息,可以选择多个锚点节点,如图 5-2(b)所示。PGNN^[12] 进一步引入锚点集的概念,如图 5-2(c)所示,通过锚点 s_1 和 s_2 无法区分节点 v_1 和 v_3 ,而通过将 s_1 和 v_3 组合为锚点集,把目标节点到锚点集中所有节点的最近距离作为相对距离,就可以正确区分 v_1 和 v_3 。

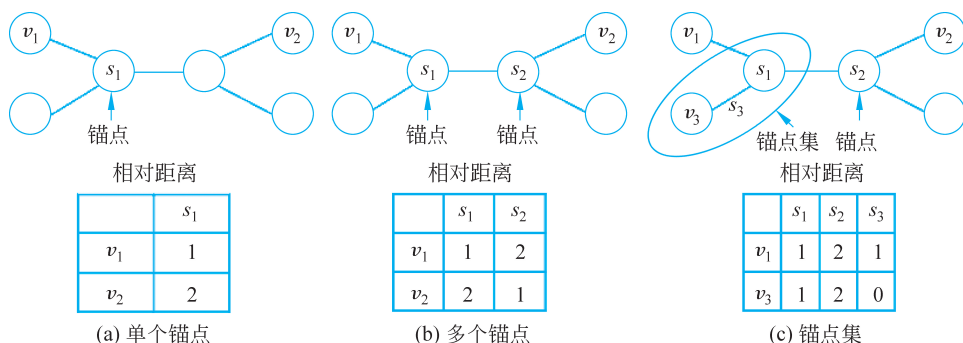


图 5-2 相对位置编码

2. 特征选择

当机器学习算法中使用的数据特征维度过高,就会在高维特征空间中呈现出稀疏性,导致需要指数级增长的数据量去维持泛化性,使模型训练的成本显著增加,这种现象被称为维度灾难。因此,特征选择旨在识别与标签高度相关的特征,并在模型训练过程中优先考虑这些特征,从而缓解维度灾难。特征选择不仅有助于降低与高维数据相关的计算成本,还通过拟合有意义的特征来提高模型泛化性能。在图学习中,常用的特征选择方法可以根据其与下游任务的关系分为两类:任务无关的特征选择和任务特定的特征选择。

1) 任务无关的特征选择

这类方法专注于选择能够适用于任何图神经网络模型或下游任务的特征,主要围绕引入正则化目标函数进行特征选择。例如,AsGNNs^[13]将正则化方法引入 GCN 和 GAT 中,将特征选择与 GNN 结合在一起,以提取有意义的特征并消除噪声特征。具体来说,以 GCN 为例,引入 $\ell_{2,1}$ 范数来对每一个图卷积层的参数进行约束,优化目标可以表示为

$$\min \mathcal{L}_{\text{gcnn}}(\mathbf{A}, \mathbf{X}; \Theta) + \sum_{k=1}^K \lambda_k \|\Theta^{(k)}\|_{2,1}$$

其中, $\|\Theta^{(k)}\|_{2,1} = \sum_{i=1}^N \sqrt{\sum_{j=1}^M |\Theta_{ij}^{(k)}|^2}$ 表示参数矩阵每一行的 2-范数总和, $\lambda_k > 0$ 是权重参数,为了简化计算,AsGNNs 将所有的 λ_k 设为 λ 。这种约束确保了学习的参数矩阵 $\Theta^{(k)}$ 具备行稀疏性,从而可以自然地进行特征选择。

2) 任务特定的特征选择

与之相对,任务特定的特征选择在进行特征选择时考虑了下游的具体任务。以 Dual-Net GNN^[14] 为例,注意到在节点分类任务中,选择性聚合的效果优于全部聚合,因此它提出利用输入节点特征的子集训练一个分类器,以预测节点标签,并且设计了一个选择器模型,学习最佳输入特征子集以实现性能提升。具体来说,分类器是一个两层的 MLP,它可以表

示为 $f_c(\theta; \mathbf{X}, m)$, 其中, θ 是网络的参数, \mathbf{X} 是节点特征矩阵, m 是指示节点特征矩阵子集的一个掩码向量; 选择器也是一个两层的 MLP, 可以表示为 $f_s(\phi, m)$, 其中 ϕ 是选择器的参数, m 同样是一个掩码向量, 选择器的输出是一个标量值, 表示输入掩码向量在分类器上的预测性能。

在训练时, 分类器采用交叉熵损失, 通过最小化选择器输出的预测性能与分类器真实的分类性能之间的均方误差来优化选择器。具体的训练流程分为三个阶段, 第一阶段在节点特征矩阵的不同组合上训练分类器和选择器。对于每次前向传播, 会随机采样一个节点特征子集作为分类器的输入。同样, 对应的掩码向量也被设定为选择器的输入。分类器和选择器使用其对应的损失函数进行优化。训练直到分类器对每种掩码组合产生稳定的损失, 且选择器能够学会根据掩码得到分类器的性能。如图 5-3 所示, 第二阶段目标是利用训练好的选择器为分类器生成可能的最优掩码。Dual-Net GNN 认为输入中具有较大梯度的成分对模型输出的贡献更大, 因此, 首先使用一个所有索引权重相等的掩码向量(如 1/2)作为输入给选择器, 然后计算相对于输入向量的梯度并挑选前 p 大梯度对应的索引。但最优子集可能是一个更小的子集, 于是从中固定采样若干组合, 并计算分类器的验证损失, 选择损失最小的掩码对分类器和选择器进行一次训练。第二阶段重复若干次后进入第三阶段, 选择验证损失最小的输入掩码, 仅对分类器继续训练, 直到收敛。

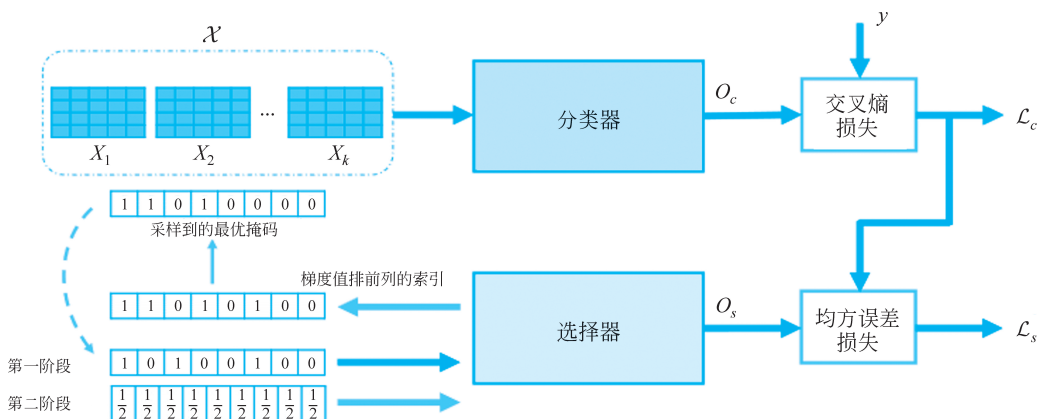


图 5-3 Dual-Net GNN 阶段二训练流程^[14]

3. 特征补全

大多数图神经网络假设图中的节点特征是完整的, 但这一假设在实际应用中往往并不能成立, 主要原因包括以下几个方面: ①数据收集过程中出现的机器或人为错误; ②收集完整数据集在实际应用中成本很高; ③许多用户由于隐私保护不愿提供完整个人信息。因此, 为了解决图中特征不完整的问题, 特征补全作为一种重要解决方案, 旨在填补图中缺失的节点特征。根据不同类型的图数据, 现有方法可分为基于同质图的特征补全和基于异质图的特征补全。

1) 基于同质图的特征补全

大多数的同质图神经网络往往假设图具有完整的特征信息, 没有针对特征缺失图进行设计, 因此无法提供令人满意的学习效果。结构-属性转换器(SAT^[15])对图提出了共享潜在空间的假设, 并开发了一种基于分布匹配的图神经网络, 用于处理特征缺失的图。