

DOI:10.12405/j.issn.2097-1486.2026.02.004

编者按:

有效处理复杂关系与非欧式空间数据一直是人工智能领域的重要目标之一。近年来,图神经网络因能直接操作图结构并利用消息传递机制高效聚合邻域信息而备受关注,在推荐系统、知识图谱等领域展现出强大性能。本文从静态建模到动态演化两大维度系统梳理了图神经网络的发展脉络与典型模型,重点探讨了动态图神经网络的演化机制;同时,客观分析了该领域面临的挑战与局限,并对未来重要研究方向进行了前瞻性展望,为从事相关研究的学者提供了宝贵的理论指导与研究思路。

图神经网络综述:从静态建模到动态演化

崔致林,郭虎升*

山西大学 计算机与信息技术学院,山西 太原 030006

摘要:图神经网络(Graph Neural Network, GNN)是一类用于处理图数据结构的深度学习模型,因其在处理非欧式空间数据和复杂特征方面的优势,受到广泛关注,并被广泛应用于推荐系统、知识图谱、交通道路分析等领域。GNN通过结合图结构中节点和边的信息,能够有效地学习图的表示。这些方法可以应用于静态图和动态图中,静态图是指一个单一且固定的图结构,而动态图会随着时间的推移发生变化。本文首先介绍了从卷积神经网络(Convolutional Neural Networks, CNN)到图神经网络(GNN)的发展历程,接着介绍了一些图表示的基本概念。然后,本文讨论了图神经网络的基本任务,并根据其消息传递机制介绍了常见的图神经网络模型。针对动态图的处理,本文根据动态性粒度将图神经网络分为离散型和连续型两类。最后,本文对图神经网络所面临的挑战及未来研究方向进行了展望。

关键词:图神经网络;动态图神经网络;消息传递机制;图卷积网络;节点分类;链路预测

中图分类号: TP183

文献标志码: A

文章编号: 2097-1486(2026)02-0141-19

A survey of Graph Neural Network: from static modelling to dynamic evolution

CUI Zhilin, GUO Husheng*

School of Computer Science and Information Technology, Shanxi University, Taiyuan 030006, Shanxi, China

Abstract: Graph Neural Network (GNN) is a deep learning model for processing graph data structures. It has garnered widespread attention due to its ability to handle non-Euclidean spatial data and complex features and has been widely applied in recommendation systems, knowledge graphs, and traffic analysis. By integrating node and edge information within graph structures, GNN effectively learn graph representations. These methods can be applied to both static and dynamic graphs: static graphs denote a single, fixed structure, while dynamic graphs undergo changes over time. This paper first outlines the evolutionary trajectory from Convolutional Neural Networks (CNN) to GNN, followed by an introduction to fundamental concepts in graph representation. Subsequently, the paper discusses fundamental tasks for GNN and introduces common models based on their message-passing mechanisms. For dynamic graph processing, GNN are categorised into discrete and

*通信作者

基金项目:山西省科技战略研究专项(202404030401039)

收稿日期:2025-09-19;修回日期:2026-02-24

continuous types according to the granularity of dynamic changes. Finally, the paper provides an outlook for future research directions.

Keywords: Graph Neural Network; Dynamic Graph Neural Networks; message-passing mechanism; Graph Convolutional Networks; node classification; link prediction

0 引言

在现代数据科学和人工智能领域,图数据结构因其能够有效表示复杂关系和非欧式空间数据的能力^[1-3],日益受到研究者的关注。传统的深度学习模型,如卷积神经网络(Convolutional Neural Networks, CNN),在处理具有规则网格结构的数据(如图像和视频)时表现出色^[4-5]。然而,许多实际应用场景中,数据通常以图的形式存在,如社交网络、知识图谱、生物分子结构以及交通网络等。这类数据的复杂性和非线性特征使得传统的神经网络模型难以直接应用于图结构数据上。为了解决这一问题,图神经网络(Graph Neural Network, GNN)^[6-7]应运而生。

图神经网络是一类专门用于处理图结构数据的深度学习模型。与传统的神经网络不同,GNN能够直接在图的节点和边上操作,从而有效捕捉图中各个元素之间的复杂关系。通过消息传递机制,GNN可以整合来自节点邻域的信息,使得图的表示能够更好地反映图结构和节点属性。由于其在处理非欧式空间数据方面的优势,GNN已经成为近年来学术界和工业界的研究热点。

GNN的应用场景极为广泛。从推荐系统^[8]中的用户行为预测,到知识图谱^[9-10]的实体关系推理,再到交通网络的拥堵预测^[11],GNN在各个领域展现出了强大的性能。这些应用不仅验证了GNN在处理复杂图数据时的优势,也进一步推动了GNN模型的快速发展。此外,GNN不仅可以处理静态图^[12-13],还可以处理动态图^[14-15]。静态图通常指结构和内容相对固定的图,而动态图则指那些随时间或事件动态变化的图。为了有效地处理动态图,研究者提出了离散型和连续型两类动态图神经网络模型,这些模型能够捕捉和预测动态图的演化过程。

尽管GNN已经在诸多领域取得了显著进展,但仍面临诸多挑战。例如,随着图的规模不断扩大,如何高效地训练和推理GNN模型成为一大难题^[16]。同时,由于图数据通常具有高度异质性^[17],如何设计具有普适性和鲁棒性的GNN模型,也是研究者关注的焦点。此外,在处理动态图时,如何准确捕捉时间维度上的变化,以及如何应对图的动态特性对

模型提出的挑战,也亟需进一步探索。

综上所述,本文将围绕图神经网络的多个方面展开讨论。首先,我们将简要回顾从卷积神经网络到图神经网络的发展历程,介绍一些基本的图表示概念。接下来,本文将详细讨论图神经网络的基本任务和常见模型,特别是基于不同消息传递机制的模型。此外,针对动态图的处理,我们将介绍离散型和连续型动态图神经网络的区别和应用场景。最后,本文将总结图神经网络在实际应用中的局限性,并对未来可能的研究方向进行展望。

通过本文的综述,希望能够为研究者提供一个全面的图神经网络领域的概览,并为未来的研究工作提供参考和启示。

1 图神经网络的发展

图神经网络(GNN)的发展历程可以追溯到图嵌入方法^[18]的早期研究,这些研究为GNN的诞生和发展奠定了重要的基础。在理解图神经网络的发展过程中,两个关键的方面是:早期的图嵌入方法以及从卷积神经网络(CNN)到图神经网络的演变。

1.1 早期图嵌入方法

图嵌入方法是图数据处理的早期尝试之一,旨在将图结构中的节点映射到低维向量空间中,同时保留图的拓扑结构和节点的语义信息。这些嵌入表示可以作为传统机器学习算法的输入,以解决图分类、节点分类、链路预测等任务。早期的图嵌入方法主要集中在以下几类:

基于矩阵分解的方法^[19-20]通过图的邻接矩阵或拉普拉斯矩阵进行分解,得到图中节点的低维嵌入表示。例如,经典的Spectral Clustering^[21]方法通过计算图拉普拉斯矩阵的特征向量来进行节点嵌入。尽管这些方法在理论上具有较好的解释性,但由于矩阵分解的计算复杂度较高,且难以处理大规模图,限制了其实际应用。

随机游走方法^[22-24]捕捉图中节点的局部和全局结构,研究者提出了基于随机游走的嵌入方法,如DeepWalk^[22]和Node2Vec^[23]。DeepWalk通过模拟随机游走生成节点序列,并利用skip-gram模型对这些序列进行训练,从而得到节点的嵌入表示。Node2Vec

则在此基础上引入了更加灵活的随机游走策略,使得嵌入能够更好地平衡图的局部和全局信息。随机游走方法相比矩阵分解方法,具有更好的扩展性和灵活性,因此在大规模图的嵌入任务中得到了广泛应用。

基于图核的方法^[25-27]试图通过定义图之间的相似性度量,将图的嵌入问题转化为核函数的计算问题。经典的图核方法包括 Shortest Path Kernel^[25]、Weisfeiler-Lehman Kernel^[26]等。这类方法在处理图分类问题时表现出色,但其计算复杂度较高,且难以处理异构图和大规模图。

上述这些早期的图嵌入方法为图数据处理提供了重要的思路和工具,但它们通常依赖于预定义的图结构特征,无法通过数据自动学习到更复杂的表示。同时,这些方法多为静态的,无法适应动态图的演变。这些限制激发了研究者探索更加灵活和自动化的图表示学习方法,从而推动了图神经网络的诞生。

1.2 从卷积神经网络到图神经网络的演变

CNN 因其在计算机视觉领域的成功而广受关注,尤其在图像分类、目标检测等任务中取得了巨大的成功。CNN 通过卷积操作提取局部特征,并通过多层卷积构建图像的层级表示。然而,CNN 的卷积操作依赖于数据的规则网格结构(如像素排列的图像),这使得其难以直接应用于非欧式空间的图数据。

为了将 CNN 的强大表示能力扩展到图结构数据,研究者提出了一系列将卷积操作推广到图结构上的方法,从而演化出图神经网络。这一演变主要经历了以下几个阶段:

谱域卷积方法^[28-32]:早期的图卷积方法主要基于谱图理论,试图将卷积操作定义在图的谱域中。代表性工作是 Bruna 等人提出的谱图卷积网络^[29],其通过图拉普拉斯矩阵的特征分解,将卷积操作定义为在谱域中的滤波器。然而,这类方法依赖于全局拉普拉斯矩阵的特征分解,计算复杂度较高,且难以泛化到不同图结构。

空间域卷积方法^[33-38]:为了克服谱域方法的限制,研究者提出了空间域的图卷积方法。这类方法直接在图的邻域空间定义卷积操作,通过聚合节点邻域的信息来构建节点表示。最具代表性的方法是 Kipf 和 Welling 提出的图卷积网络(Graph Convolutional Network, GCN)^[38]。GCN 通过近似谱图卷积,将卷积操作简化为在节点邻域中的线性变换和聚合操作,极大地提高了计算效率,并使得模型能够处理更大规模的图。

消息传递机制的引入^[39-40]:随着空间域卷积方法的发展,图神经网络逐渐演化出了一种通用的消息传递机制。在这一机制下,节点的表示通过接收和聚合来自邻居节点的信息逐层更新。GraphSAGE^[40]和 GAT(Graph Attention Networks)^[39]是其中的代表性工作。GraphSAGE 通过随机采样邻居节点,并聚合其特征,能够在大规模图上进行高效的训练和推理。而 GAT 通过引入自注意力机制,为每个邻居分配不同的重要性权重,从而增强了模型的表达能力。

动态图的处理^[15]:随着实际应用中动态图的需求增加,研究者们开始探索如何将图神经网络扩展到动态图数据上。这类工作不仅需要考虑到图结构随时间变化的动态性,还需要捕捉节点特征的时序演变。基于时间卷积、时序递归单元等技术,动态图神经网络逐渐成为图学习领域的重要研究方向。

从早期的图嵌入方法到图神经网络的演变,研究者们通过不断借鉴和改进传统深度学习方法,成功地将卷积操作推广到图结构数据上,推动了图神经网络的快速发展。如今,图神经网络不仅在学术研究中备受瞩目,也在实际应用中展现出广泛的前景。如图 1 所示,GNN 的发展从早期的图嵌入方法演进至谱域卷积,并最终在空间域消息传递框架下得以繁荣。

2 图表示及相关概念

在讨论图神经网络(GNN)之前,首先需要理解图(Graph)的基本概念及其表示方法。图是一种数学结构,用于表示对象之间的关系。它由节点(Nodes,也称为顶点,Vertices)和边(Edges)组成,节点表示对象,而边表示对象之间的关系。

2.1 图的基本概念

节点(Nodes):节点是图中的基本元素,通常表示实际问题中的实体。例如,在社交网络中,节点可以表示用户;在分子图中,节点可以表示原子。

边(Edges):边连接两个节点,表示节点之间的关系。边可以是有向的或无向的。有向边(Directed Edges)表示节点之间存在特定方向的关系,例如网页链接;而无向边(Undirected Edges)则表示节点之间的关系是对等的,如朋友关系。

图的类型又分为无向图、有向图以及加权图。

无向图(Undirected Graph):边没有方向,表示关系是双向的。如图 2 所示,社交网络中的朋友关系可以用无向图表示。

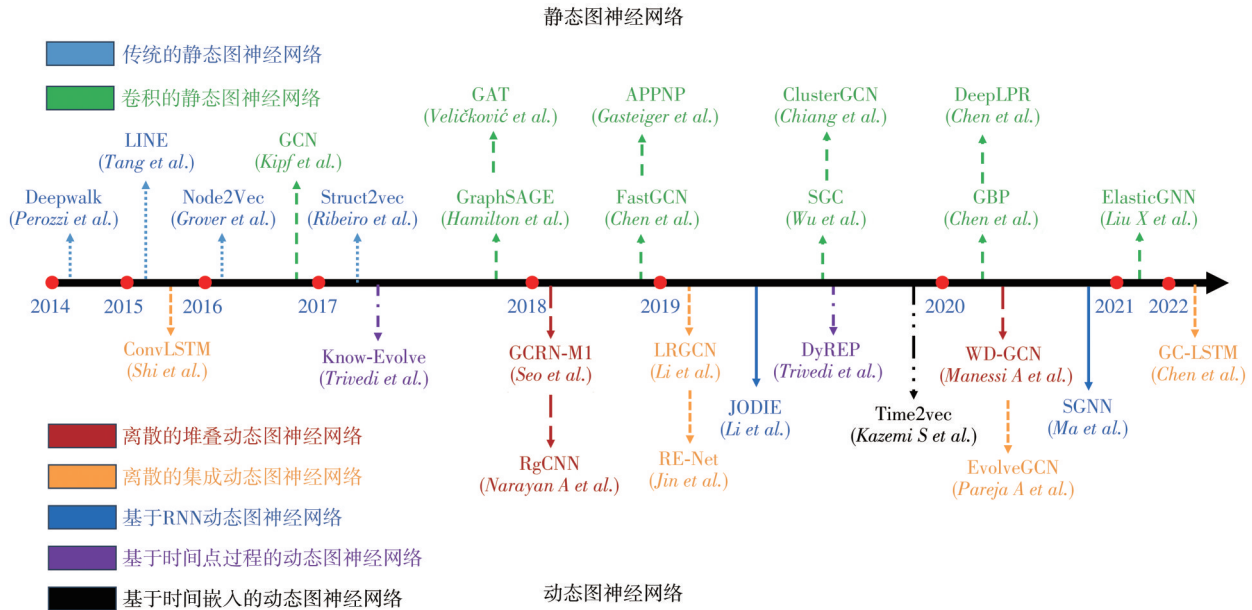


图1 发展历程

Figure 1 Development History

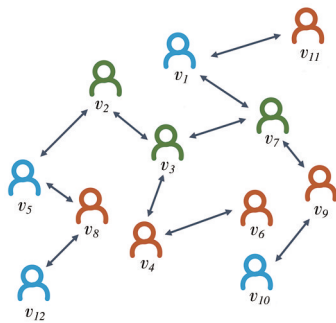


图2 社交网络

Figure 2 Social Networks

有向图(Directed Graph):边有方向,表示关系是单向的。例如,网页的超链接关系可以用有向图表示。

加权图(Weighted Graph):边附加了权重,表示关系的强度或距离。例如,交通网络中的边权重可以表示两地之间的距离或路程时间。

2.2 图的表示方法

图的表示方法主要包括邻接矩阵(Adjacency Matrix)和特征矩阵(Feature Matrix)。

邻接矩阵(Adjacency Matrix):邻接矩阵是图的一种表示方式。对于一个有 n 个节点的图,邻接矩阵 A 是一个 $n \times n$ 的矩阵,而在动态图中通常用 A^t 表示第 t 个时刻的邻接矩阵,其中 A_{ij}^t 表示从节点 i 到节点 j 的边。如果是无向图,邻接矩阵是对称的;如果是有向图,邻接矩阵不一定对称。

邻接矩阵可以表示图中节点之间的直接连接关

系,是GNN模型中的重要输入。对于稀疏图,通常会采用稀疏矩阵的存储方式,以节省空间和计算资源。

特征矩阵(Feature Matrix):特征矩阵用于描述节点的特性。对于一个有 n 个节点和 d 维特征的图,特征矩阵 X 是一个 $n \times d$ 的矩阵,而在动态图中用 X^t 表示第 t 个时刻的特征图,其中 X_i^t 表示节点 i 的特征向量。这些特征可以是节点的属性,如用户的年龄、职业,或原子的化学性质等。

在GNN中,特征矩阵与邻接矩阵共同作为输入,用于对图中的节点进行编码和表示。这种表示方法可以捕捉到图结构信息和节点特征信息,供后续的学习任务使用。这些基本概念和表示方法为GNN的后续发展奠定了基础。通过有效地表示图结构和节点特征,GNN能够在图数据上执行复杂的任务,如节点分类、链路预测和图分类等。

3 图神经网络任务

3.1 节点分类

节点分类任务(Node classification)^[41-43]通常指的是依据图中节点所具备的各类特征,例如节点自身的属性信息、与之相连的边的特征以及该节点在整个图结构中所处的具体位置,包括与其他节点的连接关系、在网络中的层次位置等要素,来对节点的类别标签进行预测。在节点分类任务中,引文网络数据集(如Cora, CiteSeer, PubMed)常用于评估模

型性能。这些数据集的特点是节点代表论文,边代表引用关系,节点特征是论文的词袋模型,任务是对论文进行主题分类。例如,GCN和GAT等模型在这些数据集上被广泛验证,展现了其利用图结构和节点特征进行半监督学习的能力。

GCN^[38]是一种用于图结构数据分析的神经网络架构(见图3),它通过在图的节点上定义卷积操作来学习节点的表示。在GCN中, A 代表图的邻接矩阵,它编码了图中节点间连接的拓扑信息; X 是节点的特征矩阵,在一些文章中GCN经常使用 H 表示节点的特征矩阵,包含了每个节点的初始特征;而 W 是图卷积层的权重矩阵,用于在特征空间中对点特征进行变换。GCN的核心公式通常表示为:

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

其中 $X^{(l)}$ 表示第 l 层的节点特征矩阵, $W^{(l)}$ 是本层的可学习的权重矩阵, $\sigma(\cdot)$ 是非线性激活函数, $\hat{A} = A + I$,其中 I 是单位矩阵, \hat{D} 是 \hat{A} 度矩阵,其对角线上的元素 $d_{ii} = \sum_j \hat{a}_{ij}$ 。

GAT^[39]是一种图神经网络,它通过引入注意力机制来对图中节点的邻居进行加权,从而允许模型更加灵活地捕捉节点间复杂的依赖关系。在GAT

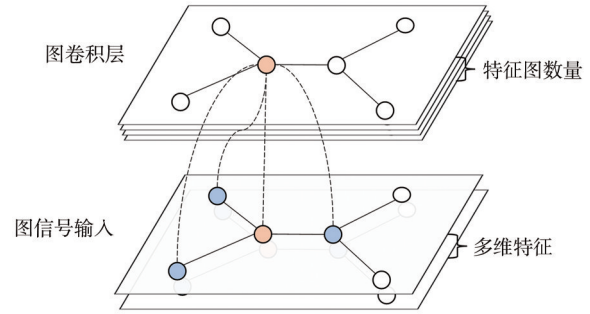


图3 GCN节点分类

Figure 3 Classification of GCN Nodes

中,每个节点利用注意力机制动态地分配不同的权重给其邻居节点,这些权重反映了邻居节点对中心节点的重要性。GAT的核心公式可以表示为:

$$h_i^{l+1} = \sigma \left(\sum_{j \in N(i)} \alpha_{ij} W^l h_j^l \right) \quad (2)$$

其中 h_i^{l+1} 是节点 i 在下一层的隐藏表示, $N(i)$ 是节点 i 的邻居节点集合, α_{ij} 是节点对邻居节点 j 的注意力权重, W^l 是当前层的权重矩阵,而 $\sigma(\cdot)$ 是激活函数。GAT的优势在于其能够自适应地学习节点间的交互,无需固定图结构或预先定义的拓扑信息,这使得GAT在节点分类任务中表现出色。如图4所示,GAT通过注意力机制为不同邻居节点赋予不同权重,以增强节点表示的区分能力。

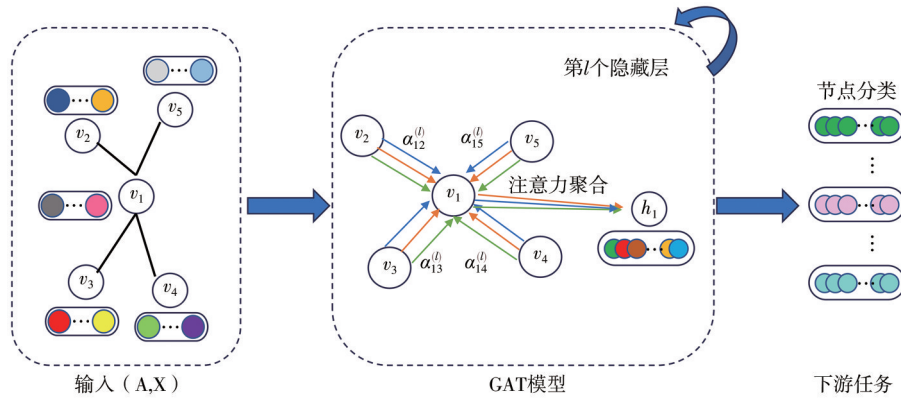


图4 GAT节点分类

Figure 4 GAT Node Classification

3.2 链路预测

链路预测^[44-46](Link Prediction)是指如何通过已知的网络节点以及网络结构等信息预测网络中尚未产生连边的两个节点之间产生链接的可能性。这种预测既包含了对未知链接的预测也包含了对未来链接的预测。

FastGCN^[47]是一种为图卷积网络设计的高效训练算法,旨在解决传统GCN训练中计算和内存效

率低下下的问题。FastGCN的核心思想是减少每层GCN中节点表示更新所需的邻居采样数量,同时保持模型性能。FastGCN通过以下步骤实现高效训练:1)节点采样:在每层GCN中,对于每个节点,FastGCN只采样一小部分邻居节点,而不是全部邻居节点,从而减少计算量。2)分层采样:FastGCN采用分层采样策略,即在不同层采样不同数量的邻居,以平衡计算效率和模型性能。3)重要性采样:

FastGCN还引入了一种基于邻居重要性的采样方法,优先选择对节点表示影响较大的邻居节点。FastGCN的节点更新公式与传统GCN相似,但考虑到采样的邻居节点集合,公式如下:

$$\mathbf{X}^{(l+1)} = \sigma \left(\sum_{i \in N_j} \frac{1}{c_{ij}} A_{ij} \mathbf{X}_i^{(l)} \mathbf{W}^{(l)} \right) \quad (3)$$

其中 $\mathbf{X}^{(l)}$ 是第 l 层所有节点的特征表示, \mathbf{A} 是邻接矩阵, $\mathbf{W}^{(l)}$ 是第 l 层的权重矩阵, N_j 是节点 j 采样的邻居节点集合, c_{ij} 是用于规范化的系数,通常与节点 j 到邻居 i 的连接强度有关, $\sigma(\cdot)$ 是激活函数。FastGCN的关键贡献在于其邻居采样策略,它通过减少每层中每个节点需要考虑的邻居数量来降低计算复杂度,同时通过分层和重要性采样保持了模型的性能。

DeepWalk^[22]是一种图嵌入算法,它通过随机游走生成捕捉节点局部结构信息的序列,并运用语言模型中的Skip-gram模型来学习这些节点的嵌入表示。在Skip-gram模型中,每个游走序列的中心节点被用来预测其周围的节点,优化目标是最大化序列中每个节点预测其前后节点的概率,公式表示为:

$$\min -\log \Pr(u_k \| \Phi(v_j)) \quad (4)$$

其中 u_k 是中心节点 v_j 周围的节点,而 $\Phi(v_j)$ 是节点 v_j 的嵌入向量。为了提高计算效率,DeepWalk采用Hi-

erarchical Softmax来近似损失函数,通过构建Huffman树减少计算量。最终,通过随机梯度下降对模型进行优化,更新节点嵌入向量以最小化损失,从而学习到能够用于节点分类或其他机器学习任务的低维连续表示。

3.3 图分类

图分类^[48-53](Graph Classification)是一种在图数据处理和分析领域中的重要任务,它指的是对于给定的一组图数据,每个图被看作一个独立的对象,且都拥有自身独特的结构、节点特征以及边的属性等信息。任务的目标是依据这些图所包含的综合特征,将它们准确地划分到预先设定好的若干个不同类别当中。对于图分类任务,其基准数据集通常由大量独立的图构成。蛋白质相互作用网络(PPI)是一个典型代表,它包含多个对应不同人体组织的子图,其任务是对蛋白质功能进行分类,这要求模型必须具备学习整个图全局表示的能力。

DGCN^[54](Dual Graph Convolutional Networks)是一种用于图基半监督分类任务的图神经网络模型,如图5所示,它通过结合两个互补的图卷积网络来提高分类性能。DGCN的核心思想是利用图的两种不同拓扑结构:节点的局部连接模式和全局连接模式,这两种模式分别由两套独立的图卷积网络来捕捉。DGCN模型的工作原理如下:

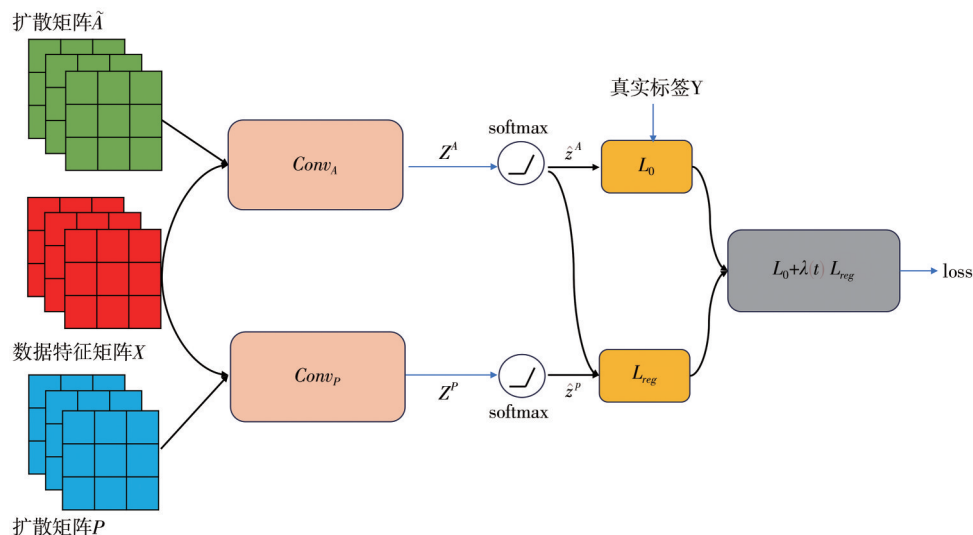


图5 DGCN图分类方法

该模型通过并行运作的局部图卷积网络和全局图卷积网络,分别学习节点的双重特征表示。最终,通过特征融合层将局部与全局信息集成,形成用于图级别分类任务的综合性图表示。

Figure 5 DGCN Graph Classification Approach

This model employs parallel operation of local graph convolutional networks and global graph convolutional networks to learn dual feature representations for nodes. Ultimately, feature fusion layers integrate local and global information to form a comprehensive graph representation for graph-level classification tasks.

1)局部图卷积网络:这个网络专注于捕捉每个节点的局部邻域信息,通常使用标准的图卷积操作来实现。对于节点 v ,其局部图卷积可以表示为:

$$H_v^{(l)} = \sigma \left(\sum_{u \in N(v)} \frac{1}{c_{uv}} A_{uv} F_u^{(l-1)} \right) \quad (5)$$

其中 $N(v)$ 是节点 v 的邻居节点集合, A 是邻接矩阵, $F_u^{(l-1)}$ 是节点 u 在第 $(l-1)$ 层的特征表示, c_{uv} 是规范化系数, $\sigma(\cdot)$ 是激活函数。

2)全局图卷积网络:这个网络旨在捕捉整个图的全局结构信息。它通过在图上定义一个全局卷积操作来实现,通常涉及对所有节点的特征进行聚合。节点 v 的全局图卷积可以表示为:

$$G_v^{(l)} = \sigma \left(\alpha \cdot F_v^{(l-1)} + \beta \cdot \frac{1}{N} \sum_{u=1}^N F_u^{(l-1)} \right) \quad (6)$$

其中 α 和 β 是平衡局部和全局特征的权重参数, N 是图中节点的总数。

3)特征融合:局部和全局图卷积网络的输出被融合以形成最终的节点特征表示。这种融合可以通过简单的堆叠或更复杂的交互方式来实现,目的是结合两种网络的优势。

4)分类层:最后,融合后的特征表示被送入一个或多个全连接层,以进行半监督分类任务。DGCN的优势在于它能够同时考虑图的局部和全局结构信息,这有助于提高模型对图数据的泛化能力和分类精度。通过这种方式,DGCN在处理具有复杂拓扑结构的图数据时,能够更好地捕捉节点之间的复杂关系,从而在各种图基分类任务中取得更好的性能。

ElasticGNN^[55]是一种先进的图神经网络(GNN)架构,它通过结合 ℓ_1 和 ℓ_2 范数的图平滑技术,显著提升了模型对局部特征的适应性。这种新颖的方法特别关注于增强GNN在处理不规则图数据时的灵活性和鲁棒性。ElasticGNN的核心是Elastic Message Passing(EMP)算法,这是一种新颖的通用消息传递方案,它不仅能够与现有的深度学习训练流程无缝集成,还提供了理论上的收敛性保证。在ElasticGNN中,图信号估计问题被重新构建,以便同时利用 ℓ_1 和 ℓ_2 范数的优势。这通过以下优化问题的形式体现:

$$\arg \min_F \left\{ \lambda_1 \|\tilde{\Delta} F\|_1 + \lambda_2 \|\tilde{\Delta} F\|_{21} + \frac{1}{2} \|F - X_{in}\|_F^2 \right\} \quad (7)$$

其中 $\tilde{\Delta}$ 是针对图结构特别设计的归一化差分算子, λ_1 和 λ_2 是用于平衡两种范数影响的正则化参数, X_{in} 代表输入的节点特征矩阵。EMP算法通过一系列精心设计的迭代步骤来求解上述优化问题,这些步

骤包括对节点特征 F 和辅助变量 Z 的更新:

$$\begin{aligned} \bar{F}_{k+1} &= F_k - \gamma \nabla f(F_k) - \gamma \tilde{\Delta}^T Z_k \\ Z_{k+1} &= \text{prox}_{\beta g^*} \left(Z_k + \beta \tilde{\Delta} \bar{F}_{k+1} \right) \\ F_{k+1} &= F_k - \gamma \nabla f(F_k) - \gamma \tilde{\Delta}^T Z_{k+1} \end{aligned} \quad (8)$$

其中 γ 和 β 是算法参数, $\gamma \nabla f(F_k)$ 是关于 F 的梯度,而 $\text{prox}_{\beta g^*}(\cdot)$ 是根据 $g^*(\cdot)$ 计算的近端算子,负责实现 ℓ_1 和 ℓ_2 范数的正则化效果。ElasticGNN的设计允许它在保持GNN强大的表征能力的同时,对图结构的微小变化更加敏感,从而在节点分类等半监督学习任务中展现出卓越的性能。此外,EMP算法的引入使ElasticGNN在面对对抗性攻击或图结构扰动时,仍能保持较高的鲁棒性。

综上所述,DGCN与ElasticGNN虽同为图分类的有效模型,但其设计哲学与适用场景存在显著差异。DGCN的核心优势在于融合局部与全局信息,适用于图的宏观拓扑结构本身即具有强判别性的任务(如社交圈子识别、特定功能的分子分类)。而ElasticGNN的核心优势在于其对图结构噪声和异常值的鲁棒性,更适用于数据质量不佳、存在对抗性干扰或需高泛化能力的场景(如真实世界传感器网络分析)。因此,在实践中,若图数据清洁且全局结构重要,可优先考虑DGCN;若对数据的鲁棒性和模型稳定性有更高要求,则ElasticGNN是更优的选择。

4 静态图神经网络

静态图神经网络(SGNNs)^[56]是专门用于处理静态图数据的图神经网络(GNN)的一个子类。在静态图中,节点和边的数量及其连接关系在一段时间内保持不变。SGNNs的核心在于通过消息传递机制更新节点特征,即使图的结构固定不变。每个节点收集来自其邻居的信息,并结合自身特征进行状态更新。这种更新过程通常涉及对邻居信息的聚合,可以采用简单的求和、平均,或者更复杂的最大池化和注意力机制。SGNNs可以设计为多层结构,每层都对节点特征进行更新,类似于图的卷积操作,从而捕捉节点的局部结构信息。这些网络能够端到端地学习,直接从图数据中提取节点或图级别的表示,用于分类、回归等任务。SGNNs在社交网络分析、分子结构预测、知识图谱和推荐系统等多个领域都有广泛的应用。典型的SGNNs架构包括图卷积网络(GCN)和图注意力网络(GAT),它们通过不同的方法更新节点特征,以识别和利用图数据中的复

杂结构和模式。

在图神经网络(GNN)^[57]的研究中,模型架构可以根据其处理图数据的方式被分为两大类:谱域方法和空域方法。谱域方法基于图的谱理论,利用图的拉普拉斯矩阵的特征分解来处理图数据,例如图卷积网络(GCN)通过特征分解实现节点特征的平滑和聚合,而图傅里叶变换^[58]则通过将图数据转换到谱域来进行信号处理和特征提取。谱域方法还包括谱聚类,它通过计算拉普拉斯矩阵的特征向量来发现图中的社区结构。另一方面,空域方法直接在图的顶点和边的原始空间上操作,不依赖于图的谱分解。图注意力网络(GAT)通过引入注意力机制动态地为每个节点的邻居分配不同的权重,实现有效的特征聚合。空域方法还包括图卷积网络的空域变体、图自编码器^[59]、图循环网络^[57](如基于LSTM^[60]或GRU^[61]的模型),以及图同构网络(GINs)^[33],这些模型通过迭代更新节点状态来学习图的表示,通常在图的原始空间上操作。每种方法都有其特定的优势和局限性,选择哪种方法通常取决于具体的应用需求和图数据的特性。

静态图神经网络模型可从计算域和架构范式两个维度进行划分:4.1与4.2小节主要依据计算域的不同,分别介绍了谱域和空域方法;而4.3至4.4小节则从架构范式的角度,依次探讨递归图神经网络、基于自编码器的图神经网络以及基于图嵌入的模型。

4.1 谱域方法

一阶ChebNet^[28]奠定了图卷积网络(GCN)的基础,并且启发了后续众多的GCN变体。论文的主要贡献可以概括为两点:1)简化计算方法:作者借鉴了Hammond等人^[62]在2011年提出的基于频谱图卷积的Wavelets on graphs via spectral graph theory,提出了一种一阶近似的简化计算方法。这种方法通过直接操作图结构数据,简化了频谱图卷积的计算过程,使得网络模型更加高效。2)半监督分类的验证:作者证明了图结构神经网络模型能够有效地处理图数据中的半监督分类问题。通过在多个公共数据集上的实验,作者展示了他们的方法在效率和准确率上与当时最先进的半监督学习方法相媲美。

该方法不仅为图神经网络的发展奠定了基础,而且通过提出一种简单而有效的图卷积层,极大地推动了图结构数据的学习和分析。下面是一阶ChebNet的详细说明:

该方法首先引入了一种一阶近似方法,在Cheb-

Net中,当 $K=1$ 并且假定拉普拉斯矩阵的最大特征值 λ_{\max} 接近2时,卷积操作可以被简化。此时,一阶ChebNet的卷积公式可以近似为:

$$x * g_{\theta} = \theta_0 x - \theta_1 D^{-\frac{1}{2}} A D^{-\frac{1}{2}} x \quad (9)$$

其中 x 是节点的特征向量, θ_0 和 θ_1 是模型参数, D 是度矩阵, A 是邻接矩阵。为了防止过拟合并减少模型复杂度,一阶ChebNet进一步假设 $\theta_0 = -\theta_1$ 。这样,图卷积操作可以进一步简化为:

$$g_{\theta} * x = \theta \left(I_N + D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \right) x \quad (10)$$

其中 I_N 是单位矩阵,由于 $I_N + D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$ 的特征值范围在 $[0, 2]$ 内,直接在神经网络中多次使用该算子可能导致数值不稳定,如梯度消失或爆炸。为了解决这个问题,论文中引入了一种重归一化技巧(normalization trick):

$$I_N + D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \rightarrow \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} \quad (11)$$

其中 $\tilde{A} = A + I_N$,并且 \tilde{D} 为 \tilde{A} 的度矩阵,即在图中为每个节点添加自环。通过重归一化技巧和激活函数,最终可以得到论文中描述的快速卷积公式:

$$H^{(l+1)} = f(H^{(l)}, A) = \sigma \left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)} \right) \quad (12)$$

其中 $H^{(l)}$ 表示第 l 层的节点特征矩阵, $W^{(l)}$ 是该层的权重矩阵, $\sigma(\cdot)$ 是激活函数。

Sardellitti等人^[63]针对有向图提出了一种新的图傅里叶变换(GFT)方法,旨在通过最小化图割大小的Lovász^[64]扩展来构建一组正交基向量,从而形成图的频域表示。这种方法首先定义了图的有向总变差(GDV),作为衡量图信号变化的度量,表达式为:

$$\text{GDV}(x) = \sum_{i,j=1}^N a_{ij} [x_i - x_j]^+ \quad (13)$$

其中 x_i 和 x_j 分别代表图中节点的特征,而 a_{ij} 表示节点 j 到 i 的有向边权重。为了找到最小化GDV的正交向量集,他们采用了两种优化算法:分裂正交性约束(SOC)方法和近端交替最小化增广拉格朗日(PAMAL)方法。SOC方法通过引入辅助变量和Bregman迭代^[65]来简化问题:

$$\min_{X, P \in \mathbb{R}^{N \times N}} \text{GDV}(X) + \frac{\beta}{2} \|X - P + B_{k-1}\|_F^2, \text{ s.t. } X = P, P^T P = I \quad (14)$$

而PAMAL方法则结合了增广拉格朗日技术和近端最小化策略,以确保算法的收敛性。此外,为了解决可能出现的生成非常稀疏向量或非常小聚类的问题,作者还考虑了最小化平衡图割大小的连续松

弛问题。通过一系列数值实验,作者们验证了他们的方法不仅能够有效地构建GFT基,而且在无监督和半监督学习任务中,如聚类和分类,能够准确地检测和利用图信号的聚类属性,从而为有向图的频域分析提供了一种新的强有力的工具。

4.2 空域方法

NN4G^[66] (Neural Network for Graphs) 是一种基于构造性前馈架构的神经网络,它允许将监督神经网络的输入域扩展到包括有向/无向、有环/无环的标记图。与传统的基于递归动态的神经网络不同,NN4G不依赖于层次数据的因果假设,而是通过构造性前馈架构和无反馈连接的神经元实现。NN4G是第一个提出的基于空间的ConvGNNs。NN4G通过直接累加节点的邻域信息来实现图的卷积。它还应用剩余连接和跳跃连接来记忆每一层的信息。因此,NN4G的下一层节点状态为:

$$\mathbf{h}_v^{(k)} = f\left(\mathbf{x}_v \mathbf{W}^{(k-1)} + \sum_{i=1}^{k-1} \sum_{u \in N(v)} \mathbf{h}_u^{(k-1)} \theta^{(k-1)}\right) \quad (15)$$

其中 $f(\cdot)$ 为激活函数, $\mathbf{h}_v^{(0)} = 0$, $\mathbf{W}^{(k-1)}$ 是前一层的参数矩阵。NN4G通过增量学习策略逐步添加并训练神经元,这意味着网络在学习过程中逐步添加新的神经元(隐藏单元)。每添加一个新神经元,就对其进行训练,然后将其输出固定,不再在后续学习中更新。一旦神经元被训练,它的状态就会被冻结,即在后续的学习过程中不再更新。这有助于简化学习过程,并防止过拟合。同时利用图遍历算法和局部上下文信息来更新状态变量,这使得网络能够捕捉图中的拓扑结构特征。该模型的输出层可以针对特定任务生成顶点级或图级的预测,适用于分类、回归等多种机器学习任务。

ST-GCN (Spatial-temporal Graph Convolutional Networks for Skeleton-based Action Recognition)^[67] 是一种先进的图卷积网络,它在处理基于骨架的动作识别时,采用了一种创新的邻居分组策略。这种策略不局限于仅考虑最短路径,而是将一个节点的邻居根据特定条件划分为多个组 Q 。每个组定义了不同的局部邻域结构,ST-GCN据此构造了 Q 个不同的邻接矩阵,每个邻接矩阵对应于一个特定的邻居组。在ST-GCN模型中,每个邻接矩阵都与一个独立的参数矩阵相结合,通过图卷积网络(GCN)进行处理。这种方法允许模型在不同的邻域组上学习到不同的特征表示。最终,这些从不同组学习到的特征被聚合(例如,通过求和)来获得每个节点的综合特

征表示。对于每个节点 v ,其特征更新可以表示为:

$$\mathbf{h}_v^{(l+1)} = \sigma\left(\sum_{q=1}^Q \mathbf{W}_q^{(l)} \cdot g_q\left(\mathbf{h}_v^{(l)}, \left\{h_u^{(l)}\right\}_{u \in N_q(v)}\right)\right) \quad (16)$$

其中, $\mathbf{h}_v^{(l)}$ 是节点 v 在第 l 层的特征, $\mathbf{W}_q^{(l)}$ 是第 q 个邻居组对应的参数矩阵, g_q 是图卷积操作, $N_q(v)$ 表示节点 v 在第 q 组的邻居。通过这种方式,ST-GCN能够更灵活地捕捉节点间的复杂关系,从而有效地提高了骨架动作识别的准确性。这种分组策略和多参数矩阵的应用,使得ST-GCN在处理具有丰富空间-时间动态特性的骨架数据时表现出色。

MPNN^[68] (Message Passing Neural Networks) 是一种图神经网络(GNN)的框架,它将图卷积视为一个信息在节点间传递的过程。在这个框架下,信息沿着图的边从一节点直接传递到另一节点,并通过迭代的方式进行多轮传递,从而实现信息的深入传播。MPNN通过定义消息传递函数来实现空间图卷积,该函数可以表示为:

$$\mathbf{h}_v^{(k)} = U_k\left(\mathbf{h}_v^{(k-1)}, \sum_{u \in N(v)} M_k\left(\mathbf{h}_v^{(k-1)}, \mathbf{h}_u^{(k-1)}, \mathbf{x}_{vu}^e\right)\right) \quad (17)$$

其中 $\mathbf{h}_v^{(k)}$ 表示节点 v 在第 k 轮迭代后的隐含状态, U_k 和 M_k 是有参数的函数,分别代表消息传递和消息更新的过程。 $N(v)$ 表示节点 v 的邻居节点集合, \mathbf{x}_{vu}^e 表示与边 (v, u) 相关的特征。初始时,节点的隐含状态设置为其原始特征 \mathbf{x}_v ,即 $\mathbf{h}_v^{(0)} = \mathbf{x}_v$ 。在经过 k 轮迭代后,每个节点的隐含状态 $\mathbf{h}_v^{(k)}$ 可以被用来进行节点级的预测任务,或者通过一个读出函数(readout function) R 来生成整个图的表示,用于图级的预测任务。读出函数可以定义为:

$$\mathbf{h}_G = R\left(\mathbf{h}_v^{(k)} | v \in G\right) \quad (18)$$

其中 \mathbf{h}_G 表示整个图的隐含表示, R 是一个有参数的函数,它综合了图中所有节点的隐含状态来生成图的全局表示。通过设计不同的 U_k 、 M_k 和 R 函数,MPNN可以覆盖多种现有的GNN模型,如GCN、GAT等,使其能够适应不同的应用场景和任务需求。这种灵活性和通用性使得MPNN成为图结构数据学习的重要工具。

GraphSAGE^[40] (Graph Sample and Aggregation) 是一种用于大规模图数据的归纳表示学习算法,如图6所示,GraphSAGE的核心思想是通过图结构的局部采样和聚合来学习节点的嵌入表示,这些嵌入可以用于各种下游任务,如节点分类、链路预测等。其引入了聚合函数来定义图卷积。这种聚合函数的核心作用是整合节点周围的信息,并且它需要

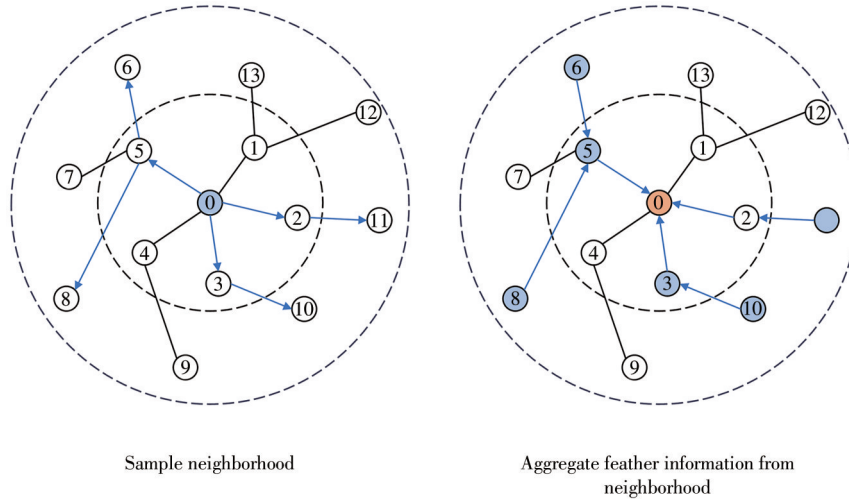


图6 GraphSAGE方法

该算法通过分层采样目标节点的固定大小邻居,并利用可微的聚合函数将采样到的邻居特征逐层聚合,从而为中心节点生成嵌入。此“采样-聚合”机制使模型能够为未见过的节点生成嵌入,实现了对大尺度图的归纳学习。

Figure 6 The GraphSAGE method

Generates embeddings for central nodes by hierarchically sampling a fixed-size neighbourhood of target nodes and aggregating the sampled neighbour features layer by layer using a differentiable aggregation function. This ‘sample-aggregate’ mechanism enables the model to generate embeddings for unseen nodes, achieving inductive learning on large-scale graphs

满足对节点顺序不敏感的特性,比如均值、求和或最大值等函数都不依赖于节点的顺序。图卷积的计算可以表示为:

$$h_v^{(k)} = \sigma\left(\mathbf{W}^{(k)} \cdot f_k\left(h_v^{(k-1)}, \{h_u^{(k-1)}, \forall u \in N(v)\}\right)\right) \quad (19)$$

其中 $h_v^{(0)} = x_v$ 表示节点的初始特征。函数 f_k 是一个聚合函数,它应该对节点的排序不敏感,例如均值、和或最大值函数。 $N(v)$ 表示节点 v 的邻居集合的一个随机采样 GraphSAGE 并没有一次性更新所有节点的状态,而是提出了一种批处理训练算法,这提高了算法在大型图上的可扩展性。GraphSAGE 的学习过程包括三个主要步骤:首先,对一个节点的 k 跳邻居进行采样;其次,通过聚合这些邻居的信息来表示中心节点的最终状态;最后,利用中心节点的最终状态进行预测和误差反向传播。

GAT^[39] (Graph Attention Networks) 是一种用于图结构数据的神经网络架构,它通过引入注意力机制来学习节点的表示。GraphSAGE 和 GCN 不同, GAT 认为相邻节点对中心节点的贡献是不同的,并且这种贡献不是预先设定的,而是通过注意力机制动态确定的。在 GAT 中,每个邻居节点对中心节点的重要性是通过注意力权重来衡量的。GAT 的图卷积操作定义如下:

$$h_v^{(k)} = \sigma\left(\sum_{u \in N(v) \cup v} \alpha_{vu} \mathbf{W}^{(k-1)} h_u^{(k-1)}\right) \quad (20)$$

其中 $h_v^{(0)} = x_v$ 表示节点的初始特征。注意力权重 α_{vu} 通过以下公式计算:

$$\alpha_{vu} = \text{softmax}\left(\mathbf{g}\left(\mathbf{a}^T \left[\mathbf{W}^{(k-1)} h_v \parallel \mathbf{W}^{(k-1)} h_u\right]\right)\right) \quad (21)$$

其中 $g(\cdot)$ 是一个激活函数, \mathbf{a}^T 是一个可学习的参数向量,而 softmax 函数确保了所有邻居节点的注意力权重之和为 1。GAT 还采用了多头注意力机制,通过并行地应用多个注意力机制并将结果进行拼接,从而增强了模型的表达能力。这种机制在节点分类任务上相较于 GraphSAGE 显示出了显著的性能提升。

PATCHY-SAN^[35] 是一种创新的图处理方法,它通过为每个节点的邻居依据特定标准排序,并与可学习的权重相结合,实现了不同位置间权重的共享。PATCHY-SAN 根据节点的图标签进行邻居排序,并选取排名靠前的 q 个邻居。图标签实际上是节点的得分,这些得分可以通过节点的度、中心性或 Weisfeiler-Lehman 颜色等多种方式获得。由于每个节点都拥有固定数量的有序邻居, PATCHY-SAN 能够将图结构数据转化为类似网格的数据结构。该方法运用标准的一维卷积滤波器来聚合节点的邻域特征,其中滤波器权值的排列与节点邻居的顺序相对应。需要注意的是, PATCHY-SAN 的排序机制仅基于图的结构特征,而非节点的特征,这通常需要进行大量的计算以处理数据。在标准的 CNN 中,通过排序函数来保持平移不变性,这对于 GCN 来说同

样重要。因此,节点的选取和排序标准在PATCHY-SAN中扮演着核心角色。在PATCHY-SAN中,排序是基于图标记进行的,这些标记考虑了图的结构特征,但并未充分考虑节点的个体特征信息。

LGCN^[41](Large-scale learnable graph convolutional networks)通过利用节点的特征信息来对节点的邻居进行排序,如图7所示,LGCN的核心思想是,对于每个节点,首先整合其邻居节点的特征矩阵,然后根据这些特征对矩阵的每一列进行排序。排序后,选择矩阵的前 k 行作为该节点的输入特征。LGCN的流程为:首先进行特征整合,对于每个节

点,将其邻居节点的特征矩阵进行整合。然后对整合后的特征矩阵按列进行排序。再从排序后的特征矩阵中选取前 k 行作为目标节点的输入数据。最后,将排序和选择后的输入数据通过一维卷积神经网络(1D-CNN)来提取目标节点的隐含特征。与PATCHY-SAN等方法相比,LGCN不需要复杂的图预处理步骤,因此更加高效。此外,为了适应大规模图数据,LGCN提出了一种子图训练策略,即将采样得到的小图作为mini-batch进行训练。这种方法使得LGCN能够有效地处理大规模图数据,同时保持了模型的学习能力。

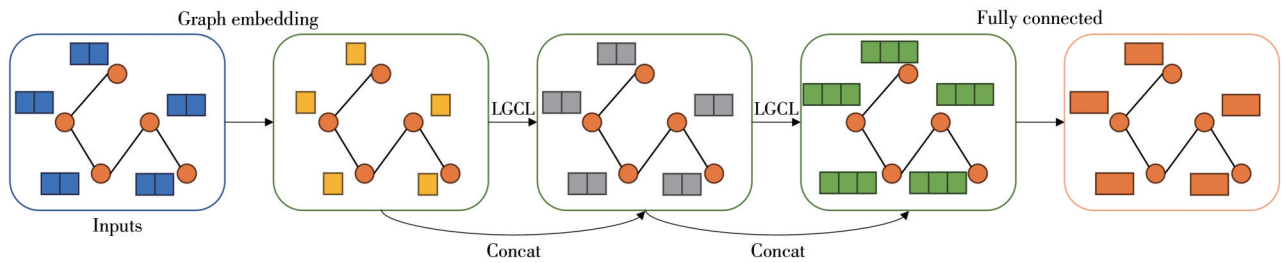


图7 LGCN方法

该模型将图结构数据转换为网格状数据。对于每个节点,它首先整合其邻居节点的特征矩阵,然后对特征矩阵的每一列(特征维度)进行排序,并选取前 k 个值构成一个规整的伪网格结构。最后,使用标准的一维卷积神经网络(1D-CNN)在此结构上学习节点的最终表示。

Figure 7 LGCN approach

Wherein the model converts graph-structured data into grid-like data. For each node, it first aggregates the feature matrices of its neighbouring nodes, then sorts each column (feature dimension) of the feature matrix and selects the top k values to form a regularised pseudo-grid structure. Finally, a standard one-dimensional convolutional neural network (1D-CNN) is employed to learn the final node representations on this structure.

在图神经网络的研究领域,模型可以根据信息传递方式的不同被分为递归图神经网络、基于自编码器的图神经网络和基于图嵌入的图神经网络。递归图神经网络通过递归方式更新节点状态,允许模型捕捉节点状态随时间的动态变化,其中 Gated Graph Neural Networks(GGNNs)^[69]通过引入门控机制来灵活处理图结构数据。基于自编码器的图神经网络,如 Graph Autoencoders(GAEs)^[70],使用编码器和解码器框架来学习图的低维表示,通过最小化原始图和重建图之间的差异来训练模型。而基于图嵌入的图神经网络,例如 Node2Vec^[23]和 DeepWalk^[22],专注于学习图中节点的嵌入表示,这些表示可以用于多种下游任务,它们通过随机游走生成节点序列,并使用词嵌入技术来学习节点的嵌入。这些方法各自具有独特的优势,适用于不同的应用场景,共同推动了图神经网络在处理图数据方面的进展。

4.3 递归图神经网络

GGNNs(Gated Graph Neural Networks)是一种在图神经网络中引入门控机制的模型,灵感来源于循环神经网络中的门控单元,如长短期记忆网络(LSTM)中的结构。GGNNs利用门控单元来精细调控信息在节点间的流动,提高了模型处理复杂图结构的灵活性。这些门控单元负责决定哪些信息被传递、哪些被抑制,从而使得模型能够捕捉节点间的复杂依赖关系。门控机制的引入不仅增强了模型对不同图结构的适应性,使其能够应对节点分类、图分类或图生成等多样化任务,还提升了模型的可解释性,因为门控操作明确显示了模型重视或忽略的特征。GGNNs的工作流程通常包括节点特征的聚合、基于节点及其邻居特征的门控信号计算、根据门控信号更新节点状态,以及这一过程在图中的递归重复,直至满足终止条件。GGNNs的应用范围广泛,从节点分类到图分类,再到图的生成,都能见到它们的

身影。总的来说,通过门控机制的引入,GGNNs在处理图数据时展现了更高的灵活性和有效性,提升了在各类图任务中的表现。

GAEs(Graph Autoencoders)是一种用于无监督学习的图神经网络,它们通过学习图数据的有效编码来重构图的节点或整个图结构。GAEs的核心思想是利用神经网络的压缩和重构过程来学习图的低维表示,这些表示可以捕捉到图中节点的拓扑结构和属性特征。GAEs通常包含两个主要部分:编码器和解码器。编码器负责将输入图映射到一个低维空间,而解码器则尝试从这个低维表示重构图的原始结构。其旨在同时学习节点的特征表示和图的结构信息,这使得它们在处理图数据时非常有效。

GAEs的工作流程通常涉及以下步骤:首先,输入图数据,包括节点特征矩阵和邻接矩阵;接着,通过编码器阶段,利用图神经网络层将每个节点的特征映射到一个低维的隐含表示空间中;然后,在解码器阶段,从这个隐含表示重建出图的结构,通常是通过预测节点间的连接概率来重构邻接矩阵;这个过程通过计算重构误差,如均方误差,来评估重构的邻接矩阵与原始邻接矩阵之间的差异;最后,利用梯度下降算法优化编码器和解码器的参数,以最小化重构误差,从而训练得到能够捕捉图数据结构和特征的低维表示。

编码器的目标是将每个节点的特征向量映射到一个较低维度的空间。这通常通过图卷积网络或其他图神经网络层来实现。设 $X \in \mathbb{R}^{N \times D}$ 为节点特征矩阵,其中 N 是节点数量, D 是每个节点的特征维度。编码器的输出 Z 可以表示为:

$$Z = \sigma(A \cdot H \cdot X) \quad (22)$$

其中 A 是邻接矩阵, H 是可学习的权重矩阵, $\sigma(\cdot)$ 是非线性激活函数。

解码器的任务是从编码器的输出 ZZ^T 重构图的邻接矩阵或节点特征。解码器通常使用一个或多个图卷积层来实现这一目标。如果我们的目标是重构邻接矩阵,解码器的输出可以表示为:

$$\hat{A} = \sigma(\tilde{A} \cdot ZZ^T) \quad (23)$$

其中 \hat{A} 是标准化的邻接矩阵(例如,通过添加自环并进行归一化处理), ZZ^T 为节点表示的外积, $\sigma(\cdot)$ 是激活函数,用于将输出限制在 $[0, 1]$ 范围内,表示边的存在概率。

通过这种方式,Graph Autoencoders能够学习到图数据的有效低维表示,这些表示可以用于多种

下游任务,如节点分类、图分类或图生成。编码器和解码器的设计使得GAEs能够捕捉到图的拓扑结构和节点特征,而损失函数则确保了重构图的质量。

4.4 基于图嵌入的图神经网络

Node2Vec^[23]是一种用于图数据的节点嵌入技术,它通过在图上进行灵活的随机游走来学习节点的表示。这种方法由Aditya Grover和Jure Leskovec在2016年的KDD(ACM SIGKDD Conference on Knowledge Discovery and Data Mining)会议上提出。Node2Vec的核心思想是将图结构数据转化为节点的低维向量表示,这些向量可以用于各种下游任务,如节点分类、链路预测和图聚类等。Node2Vec的提出者针对DeepWalk在处理带权图方面的局限性,引入了概率游走策略,并采用AliasSampling方法进行有效的采样。这项创新将在后续内容中详细讲解,目前我们关注的是它如何改进随机游走策略。在Node2Vec出现之前,DeepWalk的随机游走是完全随机的,尽管能够捕获到丰富的局部信息,但这种方法缺乏全局视角和采样策略。Node2Vec通过以下条件概率公式改进了这一点:

$$P(c_i = x | c_{i-1} = v) = \begin{cases} \frac{\pi_{vx}}{Z} & \text{if } (v, x) \in E \\ 0 & \text{otherwise} \end{cases} \quad (24)$$

其中 π_{vx} 表示未归一化的转移概率,而 Z 是归一化常数。为了使随机游走能够更好地反映网络的结构和特性,Node2Vec引入了偏置来控制游走过程。例如,最简单的偏置方法是根据边的权重选择下一个节点,此时转移概率可以表示为 $w_{vx} = \pi_{vx}$ 。然而,这种方法过于简化,因此Node2Vec引入了两个超参数 p 和 q 来控制游走策略,使得游走过程可以根据当前节点和下一个节点的最短路径距离 d_{vx} 动态调整。

$$\alpha_{pq}(t, x) = \begin{cases} \frac{1}{p} & \text{if } d_{tx} = 0 \\ 1 & \text{if } d_{tx} = 1 \\ \frac{1}{q} & \text{if } d_{tx} = 2 \end{cases} \quad (25)$$

其中超参数 p 控制着返回最近访问节点的概率,而 q 控制着向外探索的概率。这种策略使得Node2Vec能够灵活地在深度优先搜索(DFS)和广度优先搜索(BFS)之间切换,从而更全面地探索图结构。通过这种方式,Node2Vec将随机游走转化为概率游走,并通过AliasSampling算法优化了节点采样过程。

5 动态图神经网络

动态网络定义一个图 $G=(V, E)$, 其中 V 表示节点集合, 每个节点 v 都有一个对应的开始时间戳 t_s 和结束时间戳 t_e , 这些时间戳定义了节点存在的时间范围; E 表示边集合, 每条边由两个节点 u 和 v 以及它们的开始时间戳 t_s 和结束时间戳 t_e 组成, 这些时间戳定义了边存在的时间范围。这个定义突出了动态网络中节点和边随时间变化的特性, 即它们可以在特定时间范围内出现和消失。目前表示动态图的方法主要有两种: 离散时间动态图和连续时间动态图^[15]。

5.1 离散时间动态图 (Discrete Time Dynamic Graph, DTDG)

离散时间动态图^[71-76]是一种用来表示随时间变化的网络结构的数据模型。在这种模型中, 网络的动态性是通过一系列在不同时间点的快照 (snapshots) 来捕捉的。每个快照都是网络在特定时间点的静态视图, 记录了那个时刻所有节点的存在以及它们之间的连接关系。具体来说, 离散时间动态图可以表示为一系列图的集合 $G=\{G_1, G_2, \dots, G_T\}$, 其中 T 是快照的数量。每个图 G_t 包含了在时间点 t 的节点集合 V_t 和边集合 E_t 。这种表示方法允许我们使用静态网络分析方法分别对每个快照进行分析, 然后通过聚合这些快照的分析结果来获得对网络动态性的理解。离散时间动态图的这种表示方式有几个关键特点:

(1) 时间粒度: 通过选择不同的时间间隔来捕捉快照, 可以控制模型对时间变化的敏感度。

(2) 易于处理: 可以利用现有的静态网络分析工具和算法来处理每个时间点的网络快照。

(3) 信息损失: 使用时间窗口来创建快照可能会导致一些信息的丢失, 比如边的出现顺序和时间聚类信息。

EvolveGCN^[77]是一种用于动态图的图卷积网络模型, 如图8所示, 它通过递归神经网络(RNN)来演化图卷积网络(GCN)的参数, 以捕捉图序列的动态性。这种模型特别适用于处理随时间变化的图结构, 如社交网络中的友谊发展或科学引文网络中新论文的不断增长。在EvolveGCN中, 作者提出了两种架构来实现参数的演化: EvolveGCN-H和EvolveGCN-O。在EvolveGCN-H中, GCN的权重矩阵被视为动态系统的隐藏状态, 并通过门控循环单元(GRU)进行更新, 其中输入信息是节点嵌入, 而隐藏状态是

GCN权重, 更新公式为:

$$W_t^{(l)} = \text{GRU}(H_t^{(l)}, W_{t-1}^{(l)}) \quad (26)$$

其中 $W_t^{(l)}$ 表示时间步 t 和层 l 的权重矩阵, $H_t^{(l)}$ 是节点嵌入矩阵。而在EvolveGCN-O中, GCN的权重矩阵被当作动态系统的输入/输出, 使用长短期记忆网络(LSTM)来模拟权重矩阵随时间的演化, 更新公式为:

$$W_t^{(l)} = \text{LSTM}(W_{t-1}^{(l)}). \quad (27)$$

这两种方法都是在离散的时间步上进行操作的, 允许模型在每个时间点上更新其参数, 以适应图中节点和边的变化。EvolveGCN模型通过这种方式能够有效地处理节点和边的动态变化, 而不需要依赖于节点在整个时间跨度内的存在, 使其在处理新节点或节点频繁出现和消失的情况时更加灵活和有效。

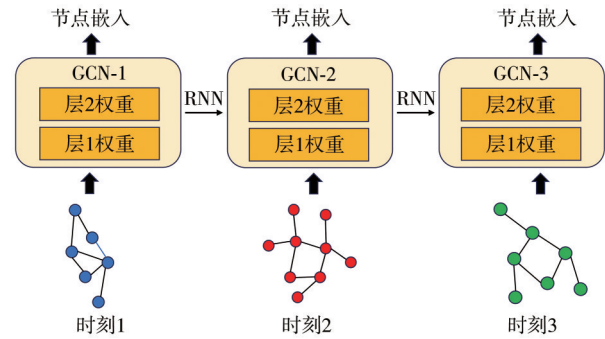


图8 EvolveGCN方法

该模型通过循环神经网络在不同时间步的图快照之间演化图卷积网络(GCN)的权重参数, 从而捕捉动态图结构的时序演化规律。

Figure 8 The EvolveGCN method

Evolves the weight parameters of a graph convolutional network (GCN) across different time steps by utilising a recurrent neural network to capture the temporal evolution patterns of dynamic graph structures.

5.2 连续时间动态图 (Continuous Time Dynamic Graph, CTCG)

连续网络^[78-81]表示是唯一具有精确时间信息的表示。这使得它们最为复杂, 但也是最具潜力的表现形式。我们涵盖了三个连续的表现形式: (1) 基于事件 (2) 接触序列和 (3) 图流。

(1) 基于事件表示: 事件驱动表示关注于图中边的活跃时间间隔, 即边在何时存在以及持续了多久。这种方法将网络中的链接视为具有时间属性的事件, 每个事件都对应一个链接的存在。网络通过一系列时间有序的事件来定义, 每个事件包含了开始时间和持续时间。公式表示为 $EB = \{(u_i, v_i, t_i, \Delta_i); i =$

$1, 2, \dots\}$, u_i 和 v_i 是发生第 i 个事件的节点对, t_i 是事件开始的时间戳。 Δ_i 是事件的持续时间。

(2) 接触序列表示: 接触序列表示将网络中的链接视为瞬时事件, 即每次链接出现时, 不考虑其持续时间。网络通过一系列时间有序的接触(即链接)事件来定义, 每个事件仅包含事件发生的时间戳和参与的节点对。公式表示为 $CS = \{(u_i, v_i, t_i); i = 1, 2, \dots\}$, 其中 u_i 和 v_i 为事件中涉及的节点对, t_i 是事件发生的时间戳。

(3) 图流表示: 图流表示将动态网络视为一个连续的边更新流, 其中每个更新都是一个事件。这种方法关注边的激活和失效, 即边的添加和删除, 而不是边的持续时间。网络通过一系列事件来定义, 每个事件包含边的两个端点、事件发生的时间戳以及一个指示边是被添加还是删除的信号。公式表示为 $GS = \{e_1, e_2, \dots\}$, 其中 $e_i = (u_i, v_i, t_i, \delta_i)$, u_i 和 v_i 是发生第 i 个事件的节点对, t_i 是事件开始的时间戳, $\delta_i \in \{-1, 1\}$ 表示边的添加(1)或删除(-1)。

DyRep^[81] 是一种用于动态图的表示学习框架, 旨在解决在动态图中有效学习节点表示的问题, 以应对现实中许多数据具有动态和复杂时间属性的挑战。DyRep 通过构建两时间尺度的深度时间点过程模型来捕捉观察到的通信和关联事件的连续时间细粒度时间动态。具体来说, 使用条件强度函数 $\lambda_k^{u,v}(t) = f_k(g_k^{u,v}(\bar{t}))$ 来建模图中事件的发生, 其中 $g_k^{u,v}(\bar{t}) = \mathbf{w}_k^T \cdot [\mathbf{z}^u(\bar{t}); \mathbf{z}^v(\bar{t})]$ 计算两个节点最近更新表示的兼容性, f_k 使用修改后的 softplus 函数 $f_k(x) = \psi_k \log(1 + \exp(x/\psi_k))$ 捕获不同时间尺度(ψ_k 表示相应过程的事件发生率)。同时, 通过一个时间关注的表示网络来学习节点表示的潜在中介过程, 该网络通过自传播、外生驱动和局部嵌入传播来更新节点表示, 具体公式为:

$$\mathbf{z}^v(t_p) = \sigma \left(\underbrace{\mathbf{W}^{struct} \mathbf{h}_{struct}^u(\bar{t}_p)}_{\text{Localized Embedding propagation}} + \underbrace{\mathbf{W}^{rec} \mathbf{z}^v(\bar{t}_p)}_{\text{Self-Propagation}} + \underbrace{\mathbf{W}^l(t_p - \bar{t}_p)}_{\text{Exogenous Drive}} \right) \quad (28)$$

此外, 提出了一种基于时间点过程的注意力机制, 通过矩阵 \mathbf{S} 计算注意力系数 $q_{ui}(t) = \frac{\exp(\mathbf{S}_{ui}(\bar{t}))}{\sum_{j \in N_u(t)} \exp(\mathbf{S}_{uj}(\bar{t}))}$, 并用于聚合信息的计算

$\mathbf{h}_{struct}^u(\bar{t}) = \max \left\{ \left\{ \sigma(q_{ui}(t) \cdot \mathbf{h}^i(\bar{t})), \forall i \in N_u(\bar{t}) \right\} \right\}$ 。在高效学习过程中, 通过最小化负对数似然 $\mathcal{L} = -\sum_{p=1}^P \log(\lambda_p(t)) + \int_0^T \Lambda(\tau) d\tau$, 来学习模型参数, 使用小批量随机梯度下降和蒙特卡罗技巧进行优化, 以实现大规模动态网络的有效处理。类似的方法还有 JODIE^[79], 如图 9 所示。

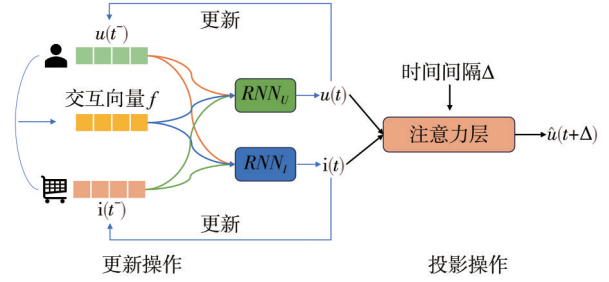


图 9 JODIE 方法

模型根据用户和项目之间随时间发生的交互事件, 利用循环神经网络动态更新其嵌入表示, 并预测用户在未来的交互轨迹或未来时刻的嵌入状态。

Figure 9 The JODIE method

Employs a recurrent network to dynamically update its embedded representations based on interaction events occurring between users and projects over time. It predicts users' future interaction trajectories or their embedded states at future time points.

综上所述, 离散时间动态图(DTDG)与连续时间动态图(CTDG)为处理动态图数据提供了两种不同范式的解决思路。离散时间动态图(DTDG)通过将时间离散化为一系列静态快照来处理动态性, 其优势在于能直接利用成熟的静态 GNN 模型、概念直观且计算高效, 尤其适用于演化规律相对缓慢、适于周期性采样的系统(如社交网络或引文网络的日度/月度分析), 但其局限性在于时间粒度的选择依赖先验知识, 且会损失快照内的精细时序信息; 而连续时间动态图(CTDG)则直接将交互建模为连续时间轴上的事件流, 能够完整保留精确的时间戳, 对高频、异步的随机事件(如金融交易、即时通讯)建模更为精确, 并支持时间插值与外推预测, 但其模型设计与计算通常更为复杂, 并面临连续时间下邻居采样和长期依赖建模的挑战。

6 结论与展望

近年来, 图神经网络(Graph Neural Networks, GNN)在处理复杂结构化数据方面取得了显著进

展,但随着应用场景的扩展和需求的变化,GNN也面临着一系列挑战。

首先,可扩展性和效率问题是GNN研究中的关键瓶颈之一。图的节点和边数在现实应用中往往非常庞大,导致计算开销急剧上升。尤其是当图的深度增加时,节点的邻域会迅速扩大,形成“邻域爆炸”问题。此外,梯度下降过程中对图的全局信息传播要求很高,这进一步增加了计算成本。因此,如何设计能够在大规模图上高效运行的GNN模型是当前的重要挑战。

其次,过平滑现象限制了深层GNN的表现力。随着网络层数的增加,节点表示逐渐趋同,导致不同节点的特征难以区分。这一现象尤其在层数较深的GNN中突出,严重影响了模型对复杂图结构的表达能力。因此,设计能够在深度网络中保持节点表示差异性的方法成为研究热点。

同时,动态图的处理也是当前GNN领域的一个重要难题。现实世界中的许多图数据是动态的,节点和边可能随时间变化。例如,在社交网络中,用户的关系和交互是随时变化的。传统的静态GNN模型在处理这类动态数据时表现不佳,因此,如何有效捕捉图的时间动态特征,开发高效的动态图建模方法是一个重要研究方向。

异构图的建模也是一个挑战。现实中的图数据往往是异构的,即节点和边具有不同的类型。如何有效处理这些异构信息,构建能够灵活适应异构图结构的GNN模型是另一个值得关注的难题。此外,GNN的可解释性也亟待提高。在许多高风险领域(如医疗、金融)中,模型的透明度和可解释性至关重要。然而,GNN的内部决策过程通常是“黑盒”,如何解释模型的预测依据是当前研究中的一个重要方向。

面对这些挑战,图神经网络领域的未来发展趋势同样充满潜力。首先,自监督学习和无监督学习的应用将得到进一步推广。由于大规模标注数据的稀缺,自监督学习通过利用图结构中的内在关系进行训练,能够在无需大量标注的情况下提高模型性能。其次,动态图建模技术的发展将会更为成熟,尤其是在时空图神经网络的研究中,将时间和空间维度结合建模动态图,提升对复杂场景的捕捉能力。

此外,多尺度和分层图学习方法将逐渐普及,这些方法通过捕捉图的局部和全局特征,在复杂图结构中表现出更强的表达能力。图强化学习也在崛起,通过结合GNN和强化学习算法,模型能够在图结构中学习最优策略,在复杂决策问题中表现出色。

最后,可解释性模型的开发将进一步推动GNN在关键领域的应用,使其更加透明和可信。

参考文献:

- [1] Zeng Yuansong, Zhou Xiang, Rao Jiahua, et al. Accurately clustering single-cell RNA-seq data by capturing structural relations between cells through graph convolutional network [C]//2020 IEEE International Conference on Bioinformatics and Biomedicine (IEEE BIBM), 2020: 519-522.
- [2] Chen Cen, Li Kenli, Teo S G, et al. Citywide traffic flow prediction based on multiple gated spatio-temporal convolutional neural networks [J]. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2020, 14(4): 1-23. DOI:10.1145/3385414.
- [3] Zehmakan A N. Majority opinion diffusion in social networks: An adversarial approach [C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2021: 5611-5619.
- [4] Chen Jianguo, Li Kenli, Bilal K, et al. A bi-layered parallel training architecture for large-scale convolutional neural networks [J]. *IEEE Transactions on Parallel and Distributed Systems*, 2018, 30(5): 965-976. DOI:10.1109/TPDS.2018.2877359.
- [5] Pu Bin, Li Kenli, Li Shengli, et al. Automatic fetal ultrasound standard plane recognition based on deep learning and IoT [J]. *IEEE Transactions on Industrial Informatics*, 2021, 17(11): 7771-7780. DOI:10.1109/TII.2021.3069470.
- [6] Gori M, Monfardini G, Scarselli F. A new model for learning in graph domains [C]//Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005, 2: 729-734.
- [7] Scarselli F, Gori M, Tsoi A C, et al. The graph neural network model [J]. *IEEE Transactions on Neural Networks*, 2008, 20(1): 61-80. DOI:10.1109/TNN.2008.2005605.
- [8] He Xiangnan, Deng Kuan, Wang Xiang, et al. Light-gcn: Simplifying and powering graph convolution network for recommendation [C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2020: 639-648.
- [9] Ji Shaoxiong, Pan Shirui, Cambria E, et al. A survey on knowledge graphs: Representation, acquisition, and applications [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2021, 33(2): 494-514. DOI:10.1109/TNNLS.2021.3070843.
- [10] Ji Guoliang, Liu Kang, He Shizhu, et al. Knowledge

- graph completion with adaptive sparse transfer matrix [C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2016: 1304-1310.
- [11] Chen Cen, Li Kenli, Teo S G, et al. Gated residual recurrent graph neural networks for traffic prediction [C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2019: 485-492.
- [12] Zhou Jie, Cui Ganqu, Hu Shengding, et al. Graph neural networks: A review of methods and applications[J]. *AI Open*, 2020, 1: 57-81. DOI: 10.1016/j.aiopen.2021.01.001.
- [13] Zhang Ziwei, Cui Peng, Zhu Wenwu. Deep learning on graphs: A survey[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2020, 34(1): 249-270. DOI: 10.1109/TKDE.2020.2981333.
- [14] Hamilton W L, Ying R, Leskovec J. Representation learning on graphs: Methods and applications[PP/OL]. *arXiv preprint* (2017-09-17) [2025-08-17]. <https://arxiv.org/abs/1709.05584>.
- [15] Skarding J, Gabrys B, Musial K. Foundations and modeling of dynamic networks using dynamic graph neural networks: A survey[J]. *IEEE Access*, 2021, 9: 79143-79168. DOI: 10.1109/ACCESS.2021.3082932.
- [16] Md V, Misra S, Ma G, et al. Distgmn: Scalable distributed training for large-scale graph neural networks[C]//Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, 2021: 1-14.
- [17] Yang Carl, Xiao Yuxin, Zhang Yu, et al. Heterogeneous network representation learning: A unified framework with survey and benchmark[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2020, 34(10): 4854-4873. DOI: 10.1109/TKDE.2020.3045924.
- [18] Goyal P, Ferrara E. Graph embedding techniques, applications, and performance: A survey[J]. *Knowledge-Based Systems*, 2018, 151: 78-94. DOI: 10.1016/j.knsys.2018.03.022.
- [19] Roweis S T, Saul L K. Nonlinear dimensionality reduction by locally linear embedding[J]. *Science*, 2000, 290(5500): 2323-2326. DOI: 10.1126/science.290.5500.2323.
- [20] Belkin M, Niyogi P. Laplacian eigenmaps and spectral techniques for embedding and clustering[C]//Advances in Neural Information Processing Systems 14. Cambridge: MIT Press, 2001: 585-591.
- [21] Von Luxburg U. A tutorial on spectral clustering[J]. *Statistics and Computing*, 2007, 17(4): 395-416. DOI: 10.1007/s11222-007-9033-z.
- [22] Perozzi B, Al-Rfou R, Skiena S. Deepwalk: Online learning of social representations[C]//Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2014: 701-710.
- [23] 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, 2016: 855-864.
- [24] Ying R, You Jiaxuan, Morris C, et al. Hierarchical graph representation learning with differentiable pooling[C]//32nd Conference on Neural Information Processing Systems (NIPS), 2018: 4805-4815.
- [25] Borgwardt K M, Kriegel H P. Shortest-path kernels on graphs [C]//Fifth IEEE International Conference on Data Mining (ICDM'05), 2005: 74-81.
- [26] Shervashidze N, Schweitzer P, Van Leeuwen E J, et al. Weisfeiler-lehman graph kernels [J]. *Journal of Machine Learning Research*, 2011, 12: 2539-2561. DOI: 10.5555/1953048.2078187.
- [27] Feng Aosong, You Chenyu, Wang Shiqiang, et al. Kergnns: Interpretable graph neural networks with graph kernels[C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2022, 36(6): 6614-6622.
- [28] Defferrard M, Bresson X, Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering[C]//Proceedings of the 30th International Conference on Neural Information Processing Systems, 2016: 3844-3852.
- [29] Bruna J, Zaremba W, Szlam A, et al. Spectral networks and deep locally connected networks on graphs[C]//2nd International Conference on Learning Representations, ICLR 2014, 2014.
- [30] Li Ruoyu, Wang Sheng, Zhu Feiyun, et al. Adaptive graph convolutional neural networks[C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2018: 3546-3553.
- [31] Levie R, Monti F, Bresson X, et al. Cayleynets: Graph convolutional neural networks with complex rational spectral filters[J]. *IEEE Transactions on Signal Processing*, 2018, 67(1): 97-109. DOI: 10.1109/TSP.2018.2879624.
- [32] Bianchi F M, Grattarola D, Livi L, et al. Graph neural networks with convolutional arma filters[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021, 44(7): 3496-3507. DOI: 10.1109/TPAMI.2021.3054830.
- [33] Xu Keyulu, Hu Weihua, Leskovec J, et al. How Powerful are Graph Neural Networks? [C]//International

- Conference on Learning Representations, 2018.
- [34] Duvenaud D K, Maclaurin D, Iparraguirre J, et al. Convolutional networks on graphs for learning molecular fingerprints [PP/OL]. *arXiv* (2015-09-30) [2025-08-17]. <https://doi.org/10.48550/arXiv.1509.09292>.
- [35] Niepert M, Ahmed M, Kutzkov K. Learning convolutional neural networks for graphs [C]//International Conference on Machine Learning, 2016: 2014-2023.
- [36] Atwood J, Towsley D. Diffusion-convolutional neural networks [C]//Advances in Neural Information Processing Systems, 2016.
- [37] Gasteiger J, Bojchevski A, Günnemann S. Predict then Propagate: Graph Neural Networks meet Personalized PageRank [C]//International Conference on Learning Representations, 2018.
- [38] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks [PP/OL]. *arXiv preprint* (2016-09-09) [2025-08-17]. <https://arxiv.org/abs/1609.02907>.
- [39] Velickovic P, Cucurull G, Casanova A, et al. Graph attention networks [C]//International Conference on Learning Representations, 2018.
- [40] Hamilton W, Ying Z, Leskovec J. Inductive representation learning on large graphs [PP/OL]. *arXiv* (2017-06-07) [2025-08-17]. <https://doi.org/10.48550/arXiv.1706.02216>.
- [41] Gao Hongyang, Wang Zhengyang, Ji Shuiwang. Large-scale learnable graph convolutional networks [C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018: 1416-1424.
- [42] Monti F, Boscaini D, Masci J, et al. Geometric deep learning on graphs and manifolds using mixture model cnns [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: 5115-5124.
- [43] Xu Keyulu, Li Chengtao, Tian Yonglong, et al. Representation learning on graphs with jumping knowledge networks [C]//International Conference on Machine Learning, 2018: 5453-5462.
- [44] Zhang Zhaoli, Li Zhifei, Liu Hai, et al. Multi-scale dynamic convolutional network for knowledge graph embedding [J]. *IEEE Transactions on Knowledge and Data Engineering*, 2020, 34(5): 2335-2347. DOI: 10.1109/TKDE.2020.3005952.
- [45] Zhang Muhan, Chen Yixin. Weisfeiler-lehman neural machine for link prediction [C]//Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017: 575-583.
- [46] Zhang Muhan, Chen Yixin. Link prediction based on graph neural networks [C]//Advances in Neural Information Processing Systems, 2018: 5165-5175.
- [47] Chen Jie, Ma Tengfei, Xiao Cao. Fastgcn: fast learning with graph convolutional networks via importance sampling [PP/OL]. *arXiv preprint* (2018-01-30) [2025-08-17]. <https://arxiv.org/abs/1801.10247>.
- [48] Gao Hongyang, Ji Shuiwang. Graph u-nets [C]//International conference on machine learning, 2019: 2083-2092.
- [49] Gilmer J, Schoenholz S S, Riley P F, et al. Neural message passing for quantum chemistry [C]//International Conference on Machine Learning, 2017: 1263-1272.
- [50] Lee J, Lee I, Kang J. Self-attention graph pooling [C]//International Conference on Machine Learning, 2019: 3734-3743.
- [51] Ma Yao, Wang Suhang, Aggarwal C C, et al. Graph convolutional networks with eigenpooling [C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019: 723-731.
- [52] Yuan Hao, Ji Shuiwang. Structpool: Structured graph pooling via conditional random fields [C]//Proceedings of the 8th International Conference on Learning Representations, 2020.
- [53] Zhang Muhan, Cui Zhicheng, Neumann M, et al. An end-to-end deep learning architecture for graph classification [C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2018: 4438-4445.
- [54] Zhuang Chenyi, Ma Qiang. Dual graph convolutional networks for graph-based semi-supervised classification [C]//Proceedings of the 2018 World Wide Web Conference, 2018: 499-508.
- [55] Liu Xiaorui, Jin Wei, Ma Yao, et al. Elastic graph neural networks [C]//International Conference on Machine Learning, 2021: 6837-6849.
- [56] 蒋玉英, 陈心雨, 李广明, 等. 图神经网络及其在图像处理领域的研究进展 [J]. *计算机工程与应用*, 2023, 59(7): 15-30. DOI: 10.3778/j.issn.1002-8331.2205-0503.
- [57] Wu Zonghan, Pan Shirui, Chen Fengwen, et al. A comprehensive survey on graph neural networks [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2020, 32(1): 4-24. DOI: 10.1109/TNNLS.2020.2978386.
- [58] Sandryhaila A, Moura J M F. Discrete signal processing on graphs: Graph fourier transform [C]//2013 IEEE

- International Conference on Acoustics, Speech and Signal Processing, 2013: 6167-6170.
- [59] Kipf T N, Welling M. Variational graph auto-encoders [PP/OL]. *arXiv preprint*(2016-11-21)[2025-08-17]. <https://arxiv.org/abs/1611.07308>.
- [60] Yu Yong, Si Xiaosheng, Hu Changhua, et al. A review of recurrent neural networks: LSTM cells and network architectures[J]. *Neural Computation*, 2019, 31(7): 1235-1270. DOI:10.1162/neco_a_01199.
- [61] Dey R, Salem F M. Gate-variants of gated recurrent unit (GRU) neural networks[C]//2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS), 2017: 1597-1600.
- [62] Hammond D K, Vandergheynst P, Gribonval R. Wavelets on graphs via spectral graph theory[J]. *Applied and Computational Harmonic Analysis*, 2011, 30(2): 129-150. DOI:10.1016/j.acha.2010.04.005.
- [63] Sardellitti S, Barbarossa S, Di Lorenzo P. On the graph Fourier transform for directed graphs[J]. *IEEE Journal of Selected Topics in Signal Processing*, 2017, 11(6): 796-811. DOI:10.1109/JSTSP.2017.2726979.
- [64] Moser R A, Tardos G. A constructive proof of the general Lovász local lemma[J]. *Journal of the ACM*, 2010, 57(2): 1-15. DOI:10.1145/1667053.1667060.
- [65] Banerjee A, Merugu S, Dhillon I S, et al. Clustering with Bregman divergences [J]. *Journal of Machine Learning Research*, 2005, 6: 1705-1749. DOI:10.5555/1046920.1194902.
- [66] Micheli A. Neural network for graphs: A contextual constructive approach[J]. *IEEE Transactions on Neural Networks*, 2009, 20(3): 498-511. DOI:10.1109/TNN.2008.2010350.
- [67] Yan Sijie, Xiong Yuanjun, Lin Dahua. Spatial temporal graph convolutional networks for skeleton-based action recognition [C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2018: 7444-7452.
- [68] Gilmer J, Schoenholz S S, Riley P F, et al. Message passing neural networks[J]. *Machine Learning Meets Quantum Physics*, 2020: 199-214. DOI: 10.1007/978-3-030-40245-7_10.
- [69] Li Yujia, Tarlow D, Brockschmidt M, et al. Gated graph sequence neural networks[PP/OL]. *arXiv preprint* (2015-11-17)[2025-08-17]. <https://arxiv.org/abs/1511.05493>.
- [70] Pan Shirui, Hu Ruiqi, Long Guodong, et al. Adversarially regularized graph autoencoder for graph embedding[PP/OL]. *arXiv preprint*(2018-02-13)[2025-08-17]. DOI:<https://arxiv.org/abs/1802.04407>.
- [71] Manessi F, Rozza A, Manzo M. Dynamic graph convolutional networks [J]. *Pattern Recognition*, 2020, 97: 107000. DOI:10.1016/j.patcog.2019.107000.
- [72] Taheri A, Gimpel K, Berger-Wolf T. Learning to represent the evolution of dynamic graphs with recurrent models[C]//Companion Proceedings of the 2019 World Wide Web Conference, 2019: 301-307.
- [73] Chen Jinyin, Wang Xueke, Xu Xuanheng. GC-LSTM: Graph convolution embedded LSTM for dynamic network link prediction [J]. *Applied Intelligence*, 2022, 52(7): 7513-7528. DOI:10.1007/s10489-021-02518-9.
- [74] Li Jia, Han Zhichao, Cheng Hong, et al. Predicting path failure in time-evolving graphs [C]//Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, 2019: 1279-1289.
- [75] Jin W, Qu Meng, Jin Xisen, et al. Recurrent Event Network: Autoregressive Structure Inference over Temporal Knowledge Graphs [C]//Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020: 6669-6683.
- [76] Bonner S, Atapour-Abarghouei A, Jackson P T, et al. Temporal neighbourhood aggregation: Predicting future links in temporal graphs via recurrent variational graph convolutions[C]//2019 IEEE international conference on big data (Big Data), 2019: 5336-5345.
- [77] Pareja A, Domeniconi G, Chen Jie, et al. Evolvegen: Evolving graph convolutional networks for dynamic graphs [C]//Proceedings of the AAAI conference on artificial intelligence, 2020, 34(4): 5363-5370.
- [78] Ma Yao, Guo Ziyi, Ren Zhaochun, et al. Streaming graph neural networks[C]//Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval, 2020: 719-728.
- [79] Kumar S, Zhang Xikun, Leskovec J. Predicting dynamic embedding trajectory in temporal interaction networks [C]//Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, 2019: 1269-1278.
- [80] Trivedi R, Dai Hanjun, Wang Yichen, et al. Know-evolve: Deep temporal reasoning for dynamic knowledge graphs [C]//International Conference on Machine Learning, 2017: 3462-3471.
- [81] Trivedi R, Farajtabar M, Biswal P, et al. Dyrep: Learning representations over dynamic graphs [C]//International Conference on Learning Representations, 2019.



崔致林,山西大学计算机与信息技术学院硕士研究生。主要研究方向为图数据挖掘等。



郭虎升,博士,教授,博士生导师,三晋英才青年优秀人才。中国计算机学会杰出会员、中国计算机学会人工智能与模式识别专委会执行委员,中国人工智能学会机器学习专委会委员、知识工程专委会委员。担任多个国际国内学术会议的出版主席、论坛主席、程序委员。近年来,主持国家自然科学基金项目2项,省部级项目10余项。曾荣获山西省科技进步二等奖、教育部宝钢教育奖、山西大学十佳青年教师、ACM理事会太原分会优博。主要研究方向为数据挖掘、机器学习等。

(责任编辑:陈 立)