

基于图注意力和改进 Transformer 的节点分类方法

李 鑫, 陆 伟, 马召祎, 朱 攀, 康 彬

(南京邮电大学物联网学院, 江苏南京 210003)

摘要: 当前, 图 Transformer 主要是在传统 Transformer 框架中附加辅助模块达到对图数据进行建模的目的. 然而, 此类方法并未改进 Transformer 原有体系结构, 数据建模精度还有待进一步提高. 基于此, 本文提出一种基于图注意力和改进 Transformer 的节点分类方法. 该方法构建基于拓扑特征增强的节点嵌入进行图结构强化学习, 并且设计基于二级掩码的多头注意力机制对节点特征进行聚合及更新, 最后引入归一前置及跳跃连接改进 Transformer 层间结构, 避免节点特征趋同引起的过平滑问题. 实验结果表明, 相较于 6 类基线模型, 该方法在不同性能指标上均可获得最优评估结果, 且能同时兼顾小规模和中规模数据集的节点分类任务, 实现分类性能全面提升.

关键词: 节点分类; 图注意力网络; Transformer; 二级掩码; 层间残差; 多头注意力

基金项目: 国家自然科学基金(No.62171232); 江苏省重点研发计划(No.BE2020729)

中图分类号: TP391 **文献标识码:** A **文章编号:** 0372-2112(2024)08-2799-12

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.12263/DZXB.20230515

A Node Classification Method Based on Graph Attention and Improved Transformer

LI Xin, LU Wei, MA Zhao-yi, ZHU Pan, KANG Bin

(School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing, Jiangsu 210003, China)

Abstract: Currently, graph Transformers mainly add auxiliary modules in the traditional Transformer framework to model graph data. However, these methods have not improved the original Transformer architecture. Their data modeling accuracy needs to be further enhanced. Thus, this paper suggests a node classification method based on graph attention and improved Transformer. In the proposed framework, a topology enhancement based node embedding is constructed for graph structure reinforcement learning. Then, a secondary mask based multi-head attention is developed for aggregation and update. Finally, pre-Norm and skip connection are introduced to improve the interlayer structure of Transformer, which can avoid the over-smoothing problem caused by feature convergence. Experimental results demonstrate that compared to 6 typical baseline models, our method is able to achieve optimal evaluation results on all different indicators. Moreover, it can simultaneously handle the node classification task for both small and medium datasets and comprehensively improve the classification performance.

Key words: node classification; graph attention network; Transformer; secondary mask; interlayer residual; multi-head attention

Foundation Item(s): National Natural Science Foundation of China (No.62171232); Key Research and Development Program of Jiangsu Province (No.BE2020729)

1 引言

近年来, 以卷积神经网络(Convolutional Neural Network, CNN)为代表的深度学习方法在机器学习领域取得了显著成效, 特别是在计算机视觉、自然语言处理等方向均取得了突破性进展^[1]. 但是, 上述成果主要面向

欧式数据, 无法处理现实世界中更多存在的非欧结构图数据, 如互联网、知识图谱、社交网络、分子结构等^[2]. 此时, 若直接将 CNN 等传统深度学习模型迁移到图数据任务中, 其表现并不尽如人意. 鉴于此, 研究人员提出图神经网络(Graph Neural Network, GNN)作为图深度

学习解决方案,并将其广泛应用于各类下游任务,如图分类^[3,4]、链路预测^[5,6]、节点分类^[7,8]等。

目前,在各类GNN方法中,如何构建节点邻居间高效的消息传递机制,一直是提高模型性能的主要关注点。例如,Kipf等人^[9]将卷积操作应用于图数据,提出了图卷积网络(Graph Convolutional Network, GCN),通过整合中心和邻居节点的特征及标签信息,为神经网络迁移至图数据提供了标准范式。随后,为了解决GCN无法区分节点重要性的问题,Velickovic等人^[10]将注意力机制引入图神经网络,提出图注意力网络(Graph Attention Network, GAT),通过学习节点邻居的不同聚合权重,自适应地传递消息及更新中心节点特征。除此之外,Wu等人^[11]还提出了SGC模型,通过连续移除激活层及预计算邻接矩阵次幂来降低模型复杂度。然而,上述模型均基于消息传递范式聚合图数据的局部特征^[12],其表达能力受W-L同构层次限制,无法学习节点间的长距离依赖关系。此外,由于模型不断重复局部聚合操作,还易出现过平滑现象^[13,14]及过挤压问题^[15]。

在此背景下,Transformer因其能捕获长距离依赖,减轻过平滑、过挤压问题受到越来越多的关注。然而,由于该模型通过全局自注意力机制将输入标记为全连接图进行处理,无法仅通过位置编码表示图结构的高度任意性。因此,目前在图表示学习任务中的表现并不尽如人意。鉴于上述原因,很多学者尝试将Transformer与图表示学习进行深度融合,对图Transformer进行了广泛研究。例如,文献[16]将GNN作为辅助模块提出了GraphTrans模型。该模型通过GNN学习节点邻域局部表示,并以位置不可知方式计算节点对影响,提高了模型的全局学习能力。而Hussain等人^[17]则采用邻接矩阵的SVD向量将其压缩成稠密位置嵌入,以此提高Transformer对图拓扑结构的捕捉能力。此外,Dwivedi等人^[18]还通过引入注意力屏蔽机制令中心节点仅关注一阶邻居,从而将全局自注意力机制改造为类似于GNN的结构,以此达到有效捕捉图拓扑信息的目的。但是,上述方法仅在Transformer中附加一些辅助模块实现图数据建模,本质上并未改变其原有体系结构,因此数据建模精度还有待进一步提高。

基于上述问题,本文提出了一种基于图注意力和改进Transformer的节点分类方法。该方法构建基于图注意力的改进Transformer模型(Graph Attention based Improved Transformer, GAITrans),首先将压缩图拓扑信息的节点位置编码融入每层输入进行图结构特征学习强化。其次,设计二级掩码矩阵,并结合组归一化、门控线性单元等优化策略,构造新的多头注意力及前馈网络对节点特征进行聚合。同时,还将层间残差结构引入Transformer框架,以跳跃连接方式进行节点特征融合。

最后,为验证GAITrans的有效性,分别在小规模及中规模数据集上进行了节点分类任务的性能分析及消融实验。实验结果表明,该模型不仅可以有效提升小规模数据集的节点分类性能,同样也可在中规模数据集上取得显著分类效果。

综上所述,本文所提基于图注意力和改进Transformer的节点分类方法的主要创新点可总结如下:

(1) 构建基于拓扑特征强化的节点嵌入模型进行图结构学习强化,以避免多层卷积后特征传播及聚合操作造成模型退化,导致模型对图结构信息学习能力不足的问题。

(2) 设计基于二级掩码的多头注意力机制对节点特征进行聚合,以避免模型梯度弥散问题,提高模型的深度建模能力。

(3) 将层间残差结构引入Transformer框架,以跳跃连接方式进行隐层节点特征融合,从而提升多层卷积后节点特征的多样性,避免特征趋同所引起的过平滑问题。

(4) 5个不同级别数据集上的实验表明,本文所提模型在Accuracy、 F_1 和Recall指标上均优于当前典型的基线模型,证明本文方法可实现不同规模数据集上节点分类性能全面提升。

2 相关工作

2.1 图注意力网络

图注意力网络是改进图卷积神经网络性能的一种有效模型^[1],其可建立目标节点与邻居节点间的不同权重关系,为研究图中节点相关性提供了理论基础。一般而言,该模型通过注意力层学习不同邻居的权重相关性,并且以卷积迭代模式聚合一阶或高阶邻居实现目标节点更新。在此,为进一步阐述图注意力网络所描述的节点权重相关性,令 $G=(V, E)$ 表示图, $V=\{v_1, v_2, \dots, v_n\}$ 、 $E=\{e_1, e_2, \dots, e_m\}$ 分别代表节点集和边集。并且,设网络卷积层数为 L , $\mathbf{h}_i^{(l)}$ 表示节点 v_i 在 l 层的特征嵌入,则 l 层的注意力计算、聚合及更新操作可表示为

$$\alpha_{i,j}^{(l)} = \text{Att}^{(l)}(\mathbf{h}_i^{(l-1)}, \{\mathbf{h}_j^{(l-1)}: j \in N(v_i)\}) \quad (1)$$

$$\mathbf{g}_i^{(l)} = \text{Agg}^{(l)}(\{\alpha_{i,j}^{(l)}, \mathbf{h}_j^{(l-1)}: j \in N(v_i)\}) \quad (2)$$

$$\mathbf{h}_i^{(l)} = \text{Update}^{(l)}(\mathbf{h}_i^{(l-1)}, \mathbf{g}_i^{(l)}) \quad (3)$$

其中, $N(v_i)$ 表示节点 v_i 的一阶或高阶邻居集合,Att、Agg、Update分别为图注意力网络的注意力计算、特征聚合及更新操作。

此时,根据上述图注意力网络模型可知,通过 L 层卷积传播后即可获得图 G 所有节点的特征嵌入,并将其用于处理节点分类、链路预测、图分类等下游任务。

2.2 Transformer 框架

Transformer 主要分为编码器和解码器两部分进行构建,并通过多层迭代实现各种下游学习任务^[19,20]. 其中,编码器由多头自注意力模块和位置前馈网络(Feed Forward Network, FFN)构成,而解码器则在编码器基础上增加了交叉注意力模块. 在此,鉴于图节点分类任务的特性,本节仅对编码器进行阐述.

此时,若令 $H = [h_1, h_2, \dots, h_n]^T \in \mathbf{R}^{n \times d}$, 表示编码器中自注意力模块的输入, h_i 为位置 i 的隐含表示,则自注意力模块可表示为

$$\text{SAtt}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \quad (4)$$

其中, $\mathbf{Q} = \mathbf{H}\mathbf{W}^Q \in \mathbf{R}^{n \times d_k}$, $\mathbf{K} = \mathbf{H}\mathbf{W}^K \in \mathbf{R}^{n \times d_k}$, $\mathbf{V} = \mathbf{H}\mathbf{W}^V \in \mathbf{R}^{n \times d}$, 为 QKV 模型的矩阵形式.

通常,Transformer 会在上述自注意力基础上引入多头注意力机制以获得更好的模型性能,其具体表达式可定义为

$$\text{Multi-Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_H)\mathbf{W}^O \quad (5)$$

其中, $\text{head}_i = \text{SAtt}(\mathbf{H}\mathbf{W}_i^Q, \mathbf{H}\mathbf{W}_i^K, \mathbf{H}\mathbf{W}_i^V)$, $\mathbf{W}_i^Q \in \mathbf{R}^{d \times d_k}$, $\mathbf{W}_i^K \in \mathbf{R}^{d \times d_k}$, $\mathbf{W}_i^V \in \mathbf{R}^{d \times d}$, $\mathbf{W}^O \in \mathbf{R}^{Hd \times d}$ 为多头映射矩阵.

随后,多头注意力输出 $H' = \text{Multi-Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$ 即可通过位置 FFN 模块进行传播. 具体而言,位置 FFN 通常设计为由 ReLU 激活层连接的全连接前馈网络,计算公式可表示如下:

$$\text{FFN}(H') = \text{ReLU}(H'\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2 \quad (6)$$

其中, $\mathbf{W}_1 \in \mathbf{R}^{d \times d_f}$, $\mathbf{W}_2 \in \mathbf{R}^{d_f \times d}$, $\mathbf{b}_1 \in \mathbf{R}^{d_f}$, $\mathbf{b}_2 \in \mathbf{R}^d$ 分别为网络映射权值和偏置.

在此,为将模型拓展至更深层次,文献[20]还在每层均引入残差连接^[21]和层归一化^[22]策略,从而得到编码器最后输出,其具体表达式为

$$H' = \text{LN}(\text{Multi-Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) + H) \quad (7)$$

$$H_{\text{out}} = \text{LN}(\text{FFN}(H') + H') \quad (8)$$

其中, H_{out} 表示最后输出, $\text{LN}(\cdot)$ 为层归一化操作.

最终,在编码器与解码器共同作用下,经过 L 层迭代后即可利用全连接网络及 softmax 分类器获得模型分类结果.

2.3 图 Transformer 框架

Transformer 不仅在多个应用领域取得了巨大成功,其在图数据建模方面也表现出了惊人潜力^[23]. 当前,图 Transformer 大多采用的编码策略是将图结构信息压缩进位置嵌入,从而使模型在具有全局推理能力的同时也能有效捕捉图数据的拓扑信息. 具体来说,位置嵌入由图邻接矩阵生成,并与节点特征融合作为模型的初始输入,对应计算公式为

$$\mathbf{X}_0 = \mathbf{X} + f_m(\mathbf{P}) \quad (9)$$

其中, \mathbf{X}_0 表示模型初始输入特征; \mathbf{X} 为图初始节点特征; $f_m(\cdot): \mathbf{R}^{d_p} \rightarrow \mathbf{R}^d$ 为位置嵌入映射; \mathbf{P} 则表示由邻接矩阵构造的位置嵌入,其可由拉普拉斯特征向量、SVD 向量等不同方式生成^[17,18,24].

随后,为了解决结构压缩所造成的信息损失问题,图 Transformer 通常还利用图拓扑信息改进式(4)中注意力矩阵的构造方法. 其中,一种较直观的方式是依据节点一阶连通性进行注意力掩码构造. 此时,式(4)可重定义为

$$\text{SAtt}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{(\mathbf{Q}\mathbf{K}^T / \sqrt{d_k}) \odot \mathbf{A}_G}{\sqrt{d_k}}\right)\mathbf{V} \quad (10)$$

其中, \mathbf{A}_G 表示图邻接矩阵.

最后,将式(10)代入传统 Transformer 框架,即可得考虑图拓扑信息的节点输出嵌入. 然而,由于该类方法并未根据图数据特性对 Transformer 模型结构进行针对性改造,模型适应性和准确性还有待进一步提升. 因此,本文针对以上问题提出了一种新的图 Transformer 模型,以期更好地建模图数据.

3 节点分类模型

根据图数据特性,本节通过融合图注意力网络并适应性改造 Transformer 框架构造一种节点分类模型 GAITrans. 该模型基于编码器优化完成,由 L 层子层堆叠组合,每层均包含改进的多头自注意力和前馈神经网络,且输入均经过拓扑学习强化,层间则通过跳跃连接进行特征融合. 最后,再通过 MLP 分类器得到分类结果. 上述模型具体的网络结构如图 1 所示,下文即对其关键组件展开说明.

3.1 基于拓扑强化的节点嵌入

根据文献[18,25],本文仍采用拉普拉斯位置编码构建基于拓扑强化的节点嵌入(Topological Reinforcement based Node Embedding, TRNE). 相应图拉普拉斯矩阵定义如下:

$$\Delta = \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2} = \mathbf{U}^T \mathbf{A} \mathbf{U} \quad (11)$$

其中, \mathbf{A} 为邻接矩阵, \mathbf{D} 为度矩阵, \mathbf{A} 为 Δ 的特征值向量, \mathbf{U} 则表示对应的特征向量矩阵.

在此,以 \mathbf{U} 中 k 个最小非平凡特征向量 $\boldsymbol{\varepsilon}_1, \boldsymbol{\varepsilon}_2, \dots, \boldsymbol{\varepsilon}_k$ 的升序构造拉普拉斯位置编码集 $\boldsymbol{\lambda}$,从而达到将图结构信息压缩入位置编码的目的. 此时, $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_{1,k}^T, \boldsymbol{\lambda}_{2,k}^T, \dots, \boldsymbol{\lambda}_{n,k}^T] = [\boldsymbol{\varepsilon}_1, \boldsymbol{\varepsilon}_2, \dots, \boldsymbol{\varepsilon}_k]$, 其中, $\boldsymbol{\lambda}_{i,k}$ 为节点 i 的拉普拉斯位置编码.

随后,将各节点位置编码与特征融合即可得考虑图结构信息的节点嵌入表示. 然而,由于节点输入特征维度 d 与位置编码维度 k 通常均不相同,因此本文采用隐空间映射对位置编码进行维度对齐. 针对节点 v ,其映射表达式可定义如下:

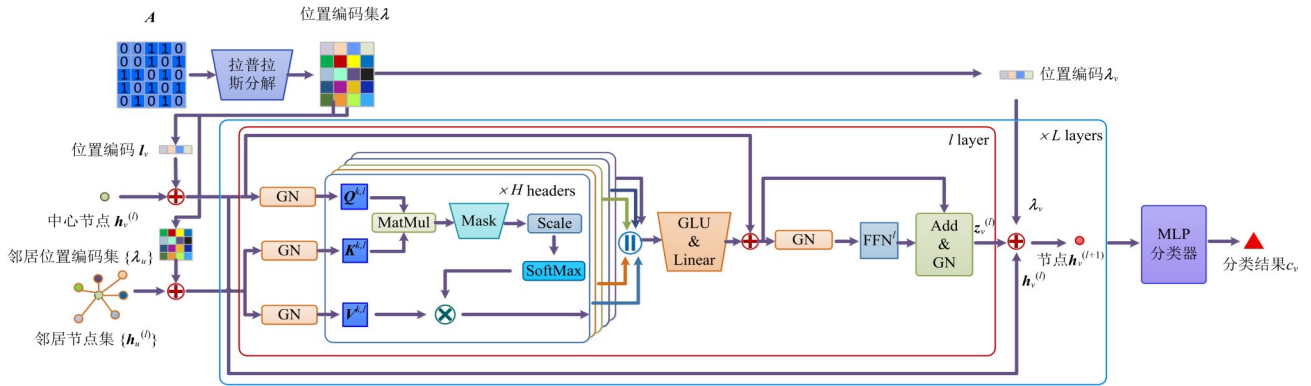


图1 节点分类模型网络结构

$$\lambda_v = W_\lambda \lambda_{v,k} + a_\lambda \quad (12)$$

其中, W_λ 、 a_λ 分别表示隐空间映射权值及偏置。

此时, 该节点在 l 层的拓扑强化特征 $h_v^{(l)}$ 可表示为

$$h_v^{(l)} = z_v^{(l)} + \lambda_v \quad (13)$$

其中, $z_v^{(l)}$ 为 l 层的节点输入特征。当 $l=0$ 时, $z_v^{(0)}$ 为节点 v 的初始输入特征, 其值可由节点 v 的初始属性 x_v 直接赋值获取, 也可通过隐空间映射获得。

在此值得注意的是, 为避免多层卷积后传播及聚合操作造成节点特征趋同, 本文在每层节点输入中均融入位置编码进行拓扑学习强化, 以提高模型对图结构信息的学习能力。

3.2 基于二级掩码的多头注意力机制

鉴于传统图 Transformer 仅通过一阶连通性构造注意力掩码矩阵, 故在此基础上, 进一步对一阶邻居节点进行二级筛选, 仅选择对分类结果有益的邻居节参与后续传播及聚合操作。在此, 将二级筛选对应的掩码矩阵定义为二级掩码矩阵 (Secondary Mark Matrix, SMM), 而基于二级掩码矩阵构造的多头注意力则定义为基于二级掩码的多头注意力机制 (Secondary Mark based Multi-Head Attention, SMMHA), 具体如图 2 所示。并且, 在注意力模块构造过程中还引入组归一化 (Group Normalization, GN)^[26] 及门控线性单元 (Gated Linear Unit, GLU)^[27] 等优化策略, 以达到提高注意力模块学习能力的目的。

此时, 针对目标节点 v , 其一阶连通性注意力掩码矩阵可由式 (10) 进一步改写为

$$A_M^v = \text{softmax} \left(\frac{h_v^{(l)} W_v^Q \cdot (h_u^{(l)} W_u^K)^T}{\sqrt{d_k}} \right) \quad (14)$$

其中, $h_u^{(l)} \in \mathbf{R}^{m \times d_k}$ 为节点 v 在 l 层的一阶邻居特征, W_v^Q 、 W_u^K 为映射矩阵。

在此, 考虑到 Transformer 结构常用的 LN 一般在每层对节点所有特征进行归一化处理, 无法捕捉局部特征间的差异性。而 GN 则允许对各特征分组学习不同的

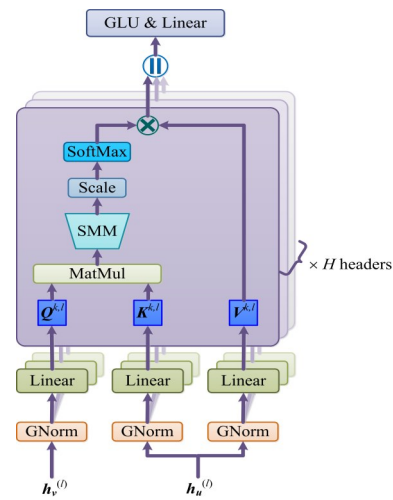


图2 基于二级掩码的多头注意力

特征分布, 有助于提高节点特征差异性的捕捉能力^[26]。故本节以 GN 代替 Transformer 的 LN 操作, 其归一化示意图如图 3 所示。

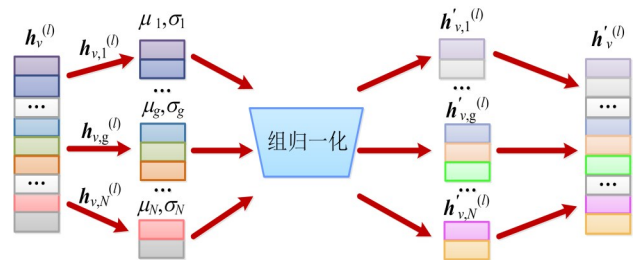


图3 组归一化示意图

由图 3 可知, GN 将节点特征进行维度分组并计算均值与方差进行归一化操作, 从而达到不受训练批大小影响的目的, 其计算方法可表示如下:

$$GN_v = GN(h_{v,g}^{(l)}) = \prod_{g=1}^N \alpha_g \frac{h_{v,g}^{(l)} - \mu_g}{\sqrt{\sigma_g^2 + \epsilon}} + \beta_g \quad (15)$$

其中, μ_g 、 σ_g 分别为组 g 特征的均值及标准差, α_g 、 β_g 则为对应归一化参数。

此时,在节点特征输入时即对其进行组归一化操作,则式(14)可进一步表示为

$$\begin{aligned} A_M^v &= \text{softmax}\left(\text{GN}_v \mathbf{W}_v^Q (\text{GN}_v \mathbf{W}_v^K)^T / \sqrt{d_k}\right) \\ &= \text{softmax}\left(\mathbf{Q}_{G,v}^{(l)} (\mathbf{K}_{G,v}^{(l)})^T / \sqrt{d_k}\right) \end{aligned} \quad (16)$$

除此之外,在特征传播及聚合过程中,部分邻居节点会因节点噪声等因素影响弱化分类效果. 基于此,本模型设计二级掩码矩阵对聚合邻居进行针对性筛选,具体构造表达式定义如下:

$$A_M^v = \text{softmax}\left(\mathbf{M}^{(l)} \mathbf{Q}_{G,v}^{(l)} (\mathbf{K}_{G,v}^{(l)})^T / \sqrt{d_k}\right) \quad (17)$$

其中, $\mathbf{M}^{(l)}$ 为 l 层二级掩码指示矩阵,其可根据任务侧重点不同选择阈值设限、注意力学习等不同构造方式. 简单起见,本文通过自注意力分数进行直接筛选判定,其示性函数为 $\mathbb{I}([\mathbf{Q}_{G,v}^{(l)} (\mathbf{K}_{G,v}^{(l)})^T]_{i,j} > 0)$.

在此,为提高模型非线性能力以避免梯度弥散,引入 GLU 操作对多头注意力输出进行激活. 此时,基于二级掩码的多头注意力机制可表示为

$$\mathbf{h}_{v,\text{cat}}^{(l)} = \parallel_{k=1}^H A_{M,k}^v \mathbf{V}_{G,v,k}^{(l)} \quad (18)$$

$$\mathbf{h}_{v,\text{GLU}}^{(l)} = (\mathbf{h}_{v,\text{cat}}^{(l)} \mathbf{W}_1^{(l)} + c^{(l)}) \odot \text{sigm}(\mathbf{h}_{v,\text{cat}}^{(l)} \mathbf{W}_2^{(l)} + d^{(l)}) \quad (19)$$

$$\mathbf{h}_{v,\text{Linear}}^{(l)} = \mathbf{h}_{v,\text{GLU}}^{(l)} \mathbf{W}_{\text{Linear}}^{(l)} \quad (20)$$

其中, $\mathbf{h}_{v,\text{cat}}^{(l)} \in \mathbf{R}^{1 \times d}$ 、 $\mathbf{h}_{v,\text{GLU}}^{(l)} \in \mathbf{R}^{1 \times d/2}$ 、 $\mathbf{h}_{v,\text{Linear}}^{(l)} \in \mathbf{R}^{1 \times d}$ 分别表示多头拼接、GLU 操作、线性映射后节点 v 的隐层特征, $\mathbf{W}_1^{(l)} \in \mathbf{R}^{d \times d/2}$ 、 $\mathbf{W}_2^{(l)} \in \mathbf{R}^{d \times d/2}$ 、 $c^{(l)}$ 、 $d^{(l)}$ 分别为 GLU 中的权值和偏置参数, $\mathbf{W}_{\text{Linear}}^{(l)} \in \mathbf{R}^{d/2 \times d}$ 为线性映射的权值参数.

3.3 基于归一前置及层内残差的 FFN

受文献[24,28,29]中 pre-Norm 结构启发,本节在保留原有 post-Norm 结构的基础上,设计增加 GN 前置操作以达到更好的优化效果,具体模型结构如图 4 所示.

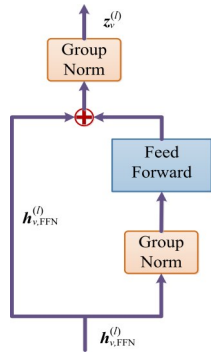


图 4 基于层内残差连接的 FFN

在此,由于上节多头注意力模块在输入时已经过 GN 处理,故本模型针对 Transformer 框架稍作变化,将多头注意力输出直接作为 FFN 模块的输入特征,其可表示为

$$\mathbf{h}_{v,\text{FFN}}^{(l)} = \mathbf{h}_{v,\text{Linear}}^{(l)} + \mathbf{h}_v^{(l)} \quad (21)$$

此时,考虑层内残差的 FFN 可定义为

$$\mathbf{z}_v^{(l)} = \text{GN}(\text{FFN}(\text{GN}_{v,\text{FFN}}) + \mathbf{h}_{v,\text{FFN}}^{(l)}) \quad (22)$$

其中, $\mathbf{z}_v^{(l)}$ 表示 FFN 模块的输出,同时也为 l 层的隐层输出特征.

3.4 基于位置编码的层间残差网络

为进一步缓解模型退化现象,避免特征趋同导致的过平滑问题,本节构造层间残差网络(Interlayer Residual Network, IRN)以保证多层卷积后节点特征的有效性. 此处,残差网络的构造方法并不唯一,可根据任务特性选择不同残差结构. 简单起见,本节以跳跃连接方式进行层间残差网络构建,随后采用 SUM Pooling 融合节点特征作为下层输入. 此时,基于跳跃残差的节点隐层特征可表示为

$$\mathbf{h}_v^{(l+1)} = \mathbf{z}_v^{(l)} + \mathbf{h}_v^{(l)} \quad (23)$$

其中, $\mathbf{h}_v^{(l+1)}$ 表示 $(l+1)$ 层节点 v 的隐层输入特征, $\mathbf{h}_v^{(l)}$ 、 $\mathbf{z}_v^{(l)}$ 分别为该节点在 l 层的隐层输入、输出特征.

此外,如 3.1 节所述,特征趋同作用会令模型对图拓扑结构的捕捉能力减弱. 因此,本节构造位置残差网络令每层均进行拓扑学习强化. 此时,加入位置编码的层间残差结构如图 5 所示.

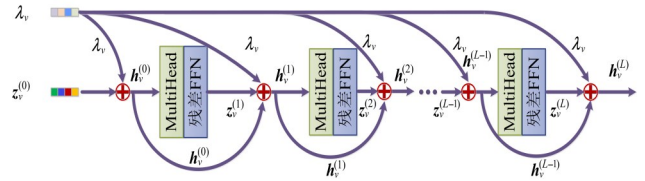


图 5 层间残差网络结构

由图 5 可知,上述层间残差网络的表达式为

$$\mathbf{h}_v^{(l+1)} = \mathbf{z}_v^{(l)} + \mathbf{h}_v^{(l)} + \lambda_v \quad (24)$$

3.5 基于节点分类任务的模型训练

由图 1 模型可知,经过上述组件处理后,利用 L 层卷积迭代即可得节点 v 的最终输出嵌入. 在此,令 \mathbf{h}_v^L 表示节点 v 的输出嵌入,则根据图 1 进行 MLP 分类后,该节点的分类预测结果可表示为

$$\hat{\mathbf{y}}_v = \text{MLP}(\mathbf{h}_v^L) = \text{softmax}(\mathbf{h}_v^L \mathbf{W}_o + \mathbf{b}_o) \quad (25)$$

其中, $\hat{\mathbf{y}}_v \in \mathbf{R}^{1 \times C}$ 表示节点 v 的最终分类预测结果, C 为分类标签数.

此时若训练集为 S ,则相应交叉熵损失函数为

$$\text{Loss} = -\frac{1}{N} \sum_{v \in S} \sum_{c=1}^C \mathbf{y}_v[c] \log \hat{\mathbf{y}}_v[c] \quad (26)$$

其中, $N = |S|$, 为训练集样本个数, $\mathbf{y}_v[c]$ 、 $\hat{\mathbf{y}}_v[c]$ 分别表示节点 v 的 one-hot 真实标签及预测标签.

4 实验结果与分析

本文在半监督实验环境中对所提模型进行性能验

证及评估. 不失一般性, 本节主要针对其节点分类任务进行性能比较和消融实验, 以此说明 GATrans 模型的有效性. 而链路预测、图分类等其他下游任务, 其建模和分析方法均与本文方法类似, 在此则不再赘述.

4.1 实验设置

4.1.1 数据集

为保证算法验证实验的全面性, 本节选取小规模

数据集 Cora、Citeseer、Pubmed 及中等规模数据集 Pattern、Cluster 作为实验数据集. 表 1 给出了各数据集的详细参数, 并分别列出了训练集、验证集及测试集的具体设定. 在此, 为保证实验客观性, 本文所有训练集、验证集及测试集划分均根据相应数据集出处的设置策略选取, 下文所有实验均在此设置上进行学习和分类预测.

表 1 实验数据集详细参数及设置

数据集	节点	边	类别	特征	标签率/%	训练集	验证集	测试集
Cora ^[30]	2 708	5 429	7	1 443	5.2	140	500	1 000
Citeseer ^[30]	3 327	4 732	6	3 703	3.6	120	500	1 000
Pubmed ^[30]	19 717	44 338	3	500	0.3	60	500	1 000
Cluster ^[25]	1 406 436	51 620 640	6	7	16.7	10 000	1 000	1 000
Pattern ^[25]	1 664 491	66 488 100	2	3	20	10 000	2 000	2 000

4.1.2 基线模型

为充分验证所提模型的节点分类效果, 本节选取 6 个典型基线模型进行对比实验, 以便从多角度说明本文方法的有效性.

(1) GCN (Graph Convolutional Network)^[9]: 该模型通过卷积方式提取特征, 并结合目标及邻居节点学习相应的嵌入表示.

(2) GraphSAGE (Graph SAmple and aggregatE)^[31]: 该模型对 GCN 进行改进, 将节点自身上层特性纳入更新方程, 并通过局部邻域采样和特征聚合生成嵌入函数.

(3) GAT (Graph Attention neTwork)^[10]: GAT 是经典的注意力模型, 采用图注意力机制计算节点间关联度, 并有区别地聚合邻域节点信息.

(4) GatedGCN (Gated Graph ConvNet)^[32]: 该模型是各向异性的 GCN 变体, 融合了残差连接、批归一化和边门控等组件.

(5) GIN (Graph Isomorphism Network)^[33]: GIN 将 W-L 同构检验进行推广, 提出深度多集理论对通用多集函数进行参数化, 从而提高 GNN 的表达能力.

(6) GT (Graph Transformer)^[18]: GT 将 Transformer 拓展到图上, 采用图拉普拉斯矩阵构造位置编码, 并引入批归一化代替了传统的层归一化.

4.1.3 参数设置

本节所有实验均基于表 2 所示实验环境进行, 模型构建在 Pytorch 及 DGL 框架上完成. 在此, 为更好地与基线模型进行对比, 本文选取 Accuracy、 F_1 和 Recall 作为模型性能的评估指标.

此外, 在模型训练时, 本实验采用 Adam 优化器进行模型优化, 并且将多头个数设置为 8, 初始学习率和衰减因子分别设置为 0.000 5、0.5. 同时, 在小规模数据集 Cora、Citeseer 和 Pubmed 上将 dropout 设置为 0.1, lay-

ers 设为 10, 而针对中规模数据集 Cluster、Pattern, 则将其 dropout 分别设为 0.1、0.01, layers 分别设为 13、12.

表 2 实验环境参数

	环境设置
OS	Ubuntu 18.04.5 LTS
RAM	256 GB DDR4@2933MT/s
CPU	Intel(R) Xeon(R) Silver 4214 CPU@2.20 GHz
GPU	GeForce RTX 2080Ti
HDD	10 TB
Software	Python 3.8.3

4.2 性能验证实验分析

4.2.1 模型超参数分析

为验证上节超参数选取对模型性能的影响, 本实验以 dropout 及 layers 为例在不同参数取值下进行性能对比实验. 图 6、图 7 分别给出了本文方法在不同数据集上超参变化对 3 类评估指标的影响曲线. 由图 6 可知, 随着 dropout 取值的增大, 各评估指标总体呈现先升后降的趋势. 在 Cora、Citeseer、Pubmed 和 Cluster 数据集上, 当 dropout 取值为 0.1 时各评估指标达到最优, 分别为: Cora# {Acc: 84.25%, F_1 : 0.844 5, Recall: 84.33%}、Citeseer# {Acc: 73.44%, F_1 : 0.739 5, Recall: 73.52%}、Pubmed# {Acc: 80.40%, F_1 : 0.805 2, Recall: 80.47%}、Cluster# {Acc: 77.33%, F_1 : 0.775 1, Recall: 77.62%}. 而在 Pattern 数据集上, 当 dropout 值为 0.001 时各评估指标即达到最优, 具体为 Pubmed# {Acc: 86.21%, F_1 : 0.874 2, Recall: 86.41%}. 因此, 本节在 Cora、Citeseer、Pubmed、Cluster 及 Pattern 数据集上将 dropout 分别设置为 0.1、0.001 进行模型性能比较实验. 此外, 由图 7 可知, 随着 layers 增大各评估指标也呈现初期提升后期下降的趋势. 具体而言, 针对 Cora、Citeseer、Pubmed 数据集, 当 layers 取值为 10 时, 各评估指标达到最优. 而对于 Clus-

ter和Pattern数据集,则在layers取值分别为13、12时达到最优.因此,本节分别将上述最优值作为layers取值进行模型性能验证实验.

4.2.2 性能对比及分析

在前文所述超参数设置基础上,本节所有实验均以Pytorch及DGL完成,并根据4.1.2节所述基线模型进行算法性能比较.在此,为验证本文方法在节点分类任

务上的有效性,本实验按小规模和中规模两种数据集情况进行性能实验,并将GATTrans与各基线模型在Accuracy、 F_1 、Recall评估指标上进行对比分析.表3给出了本文方法与各基线模型在3类小规模数据集Cora、Citeseer及Pubmed上的评估指标比较结果,其中最优化性能指标以粗体表示,而6类基线模型中的最好性能则以下划线进行标识.

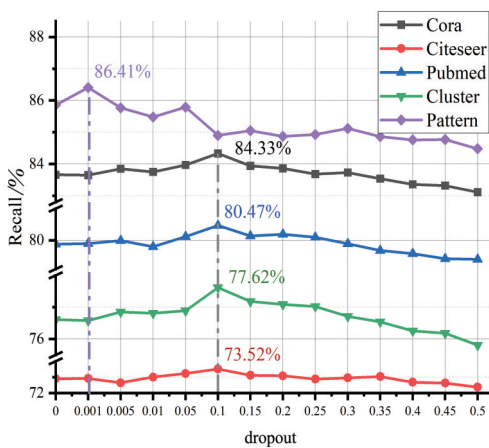
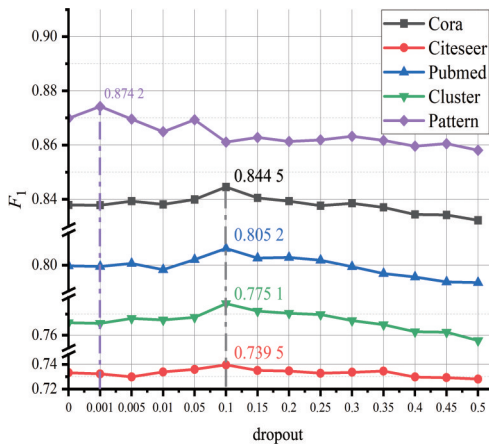
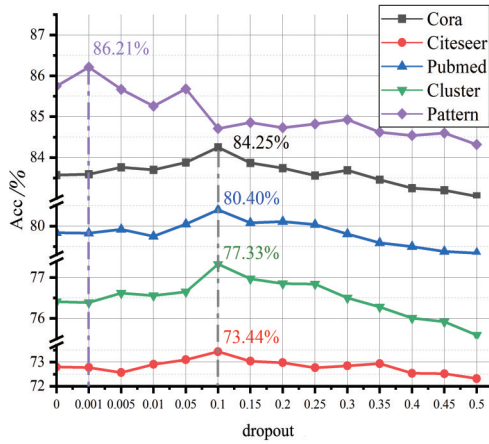


图6 dropout变化对模型评估指标的影响比较

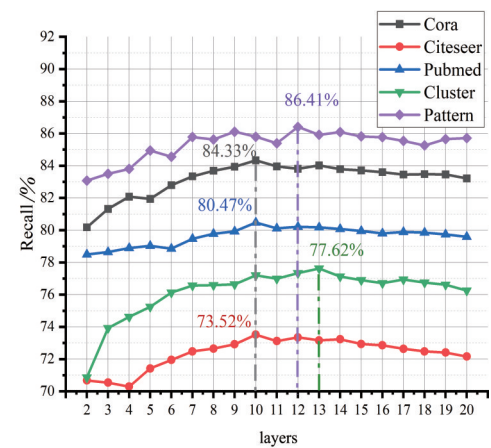
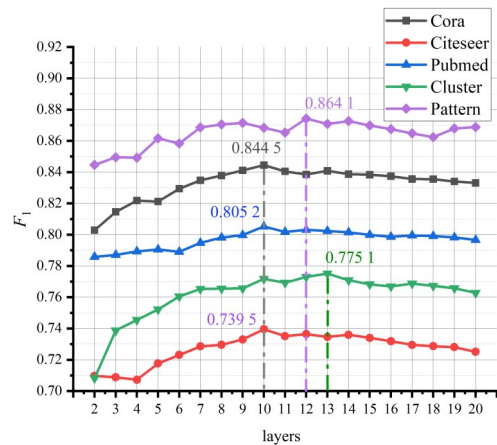
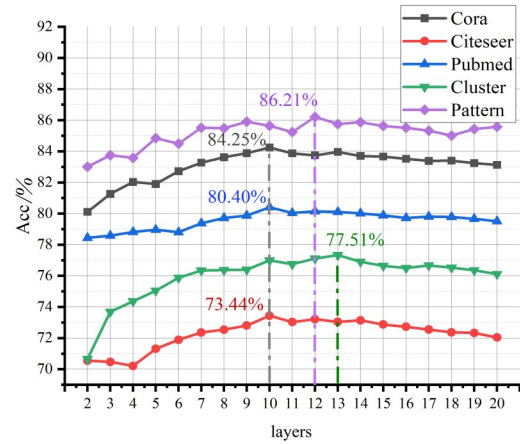


图7 layers变化对模型评估指标的影响比较

表3 本文方法与各基线模型在不同小规模数据集上的性能比较

模型	Cora			Citeseer			Pubmed		
	Acc/%	F_1	Recall/%	Acc/%	F_1	Recall/%	Acc/%	F_1	Recall/%
GCN	81.10	0.812 8	81.10	70.60	0.710 8	70.62	<u>78.90</u>	0.789 2	<u>78.94</u>
GraphSage	82.20	0.823 7	82.23	71.10	0.716 7	71.10	78.70	0.786 8	78.72
GAT	<u>82.60</u>	<u>0.827 7</u>	<u>82.60</u>	<u>72.30</u>	<u>0.728 5</u>	<u>72.34</u>	78.90	<u>0.789 4</u>	78.91
GatedGCN	80.75	0.809 6	80.74	69.83	0.702 4	69.89	76.73	0.767 7	76.74
GIN	74.40	0.746 1	74.40	64.10	0.645 5	64.10	72.70	0.726 3	72.72
GT	80.26	0.804 9	80.31	69.26	0.697 2	69.30	76.35	0.763 9	76.38
GATrans	84.25	0.844 5	84.33	73.44	0.739 5	73.52	80.40	0.805 2	80.47

由表3数据可知,本文方法在不同小规模数据集上均可获得最优性能,其与6类基线模型的最好结果相比还分别提升了 Cora# {Acc: 1.65%, F_1 : 1.68%, Recall: 1.73%}、Citeseer# {Acc: 1.14%, F_1 : 0.70%, Recall: 1.18%} 及 Pubmed# {Acc: 1.50%, F_1 : 1.58%, Recall: 1.53%}, 详细对比结果如图8所示. 究其原因, GATrans 在各层均融入位置编码进行结构特征学习强化, 可有效学习图的拓扑先验信息. 此外, 通过二级掩码矩阵设计, 也可进一步对聚合邻居进行针对性筛选, 提高模型的表征能力. 因此, 该模型相较于上述基线模型能够更好地处理小规模数据集相关下游任务.

此外, 为验证本文方法在中规模数据集上同样具有较高的有效性, 表4还给出了针对 Cluster 及 Pattern 数据集的不同模型评估比较结果. 与上文相同, 该表中最优性能指标以粗体表示, 6类基线模型的最好性能以下划线标识. 根据表4数据, GATrans 对于中规模数据集同样适用, 仍可取得令人满意的节点分类效果. 然而, 相较于本文方法, GCN、GraphSage、GAT等普遍用于处理小规模数据集的传统基线模型则表现不佳, 在中规模数据处理上出现了明显的性能下降. 而且, GIN模型针对不同数据集也呈现出不稳定状态, 特别是在 Cluster 数据集上各评估指标明显低于 GatedGCN 及 GT 模型. 反观本文方法, 其在 Cluster 及 Pattern 数据集上均取得了各评估指标的最优性能, 相较于各基线模型最优结果, 分别提升了 Cluster# {Acc: 3.54%, F_1 : 3.53%, Recall: 3.64%}、Pattern# {Acc: 0.69%, F_1 : 0.67%, Recall: 0.70%}, 详细对比结果如图9所示. 究其原因, 除了上述结构特征学习强化及基于二级掩码的多头注意力机制外, 层间残差设计也可有效缓解模型特征趋同现象, 从而使该模型能够进行更深层次的卷积处理. 因此, 相较于传统基线模型, 该模型具有更好、更全面的节点分类性能.

最后, 为进一步说明本文方法的优势及有效性, 本节对模型训练过程中不同训练次数下的分类损失进行对比实验. 图10给出了不同数据集下 GATrans 与各基线模型的收敛速度及 Loss 损失比较曲线. 由图10可知, 本文方法具有更快的收敛速度, 而且最终可以收敛

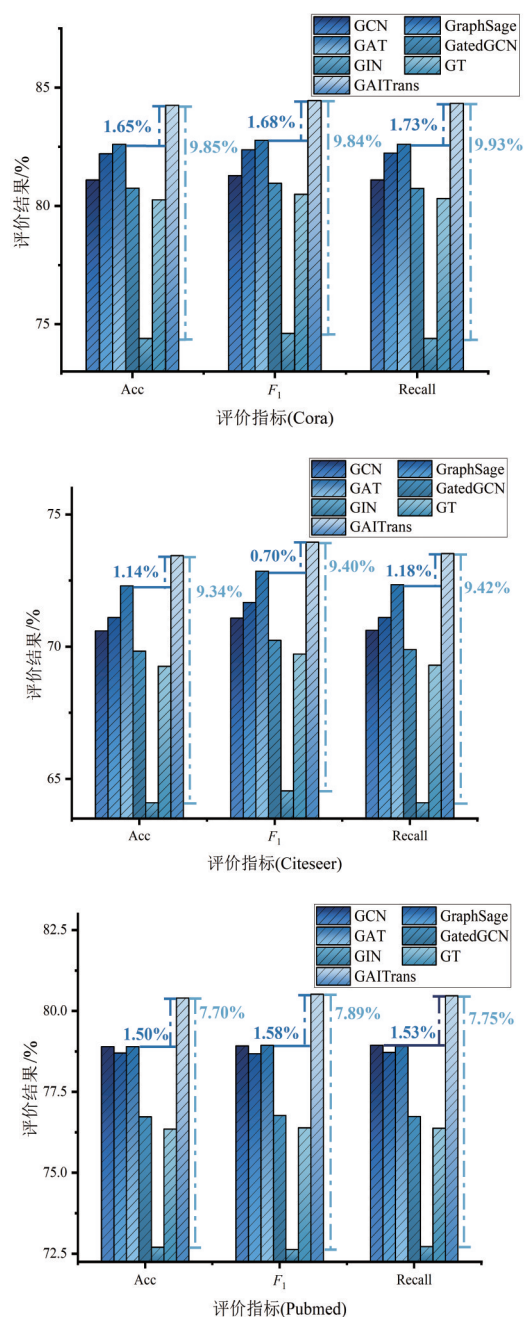


图8 Cora、Citeseer及Pubmed数据集上各指标对比结果

表4 本文方法与各基线模型在不同中规模数据集上的性能比较

模型	Cluster			Pattern		
	Acc 1%	F_1 1%	Recall 1%	Acc 1%	F_1 1%	Recall 1%
GCN	68.45	0.686 4	68.62	71.78	0.729 4	71.95
GraphSage	63.85	0.640 7	64.10	50.38	0.509 7	50.42
GAT	70.56	0.707 5	70.77	78.19	0.792 3	78.34
GatedGCN	<u>73.79</u>	<u>0.739 8</u>	<u>73.92</u>	<u>85.53</u>	<u>0.867 5</u>	<u>85.71</u>
GIN	64.73	0.649 0	64.94	85.31	0.865 2	85.45
GT	73.18	0.733 0	73.32	84.67	0.857 2	84.85
GATrans	77.33	0.775 1	77.62	86.22	0.874 2	86.41

到更低的分类损失. 因此,相较于其他基线模型,本文所提模型具有更好的收敛性及有效性.

综上所述,本文方法的综合性能优于各基线模型,其中、小规模数据集上均可获得最佳处理效果. 上述实验结果也从不同角度均验证了本文方法的性能有效性.

4.3 消融实验

为验证 TRNE、GN、GLU、SMM、IRN 等组件对模型性能的影响,本节选取 Cora、Cluster 及 Pattern 数据集进行消融实验. 通过在 GATrans 基础上移除不同组件构造多种变体进行对比验证,具体实验结果如表 5 所示. 由表中 1~6 行数据可知,分别单独添加 TRNE、

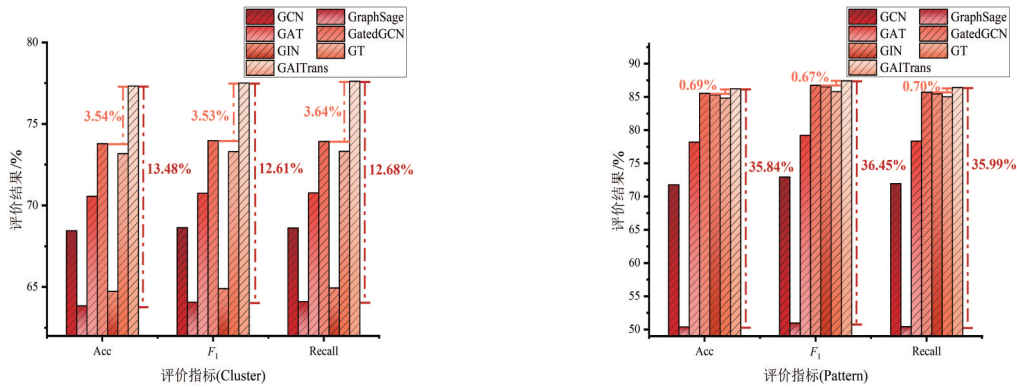


图9 Cluster 及 Pattern 数据集上各指标对比结果

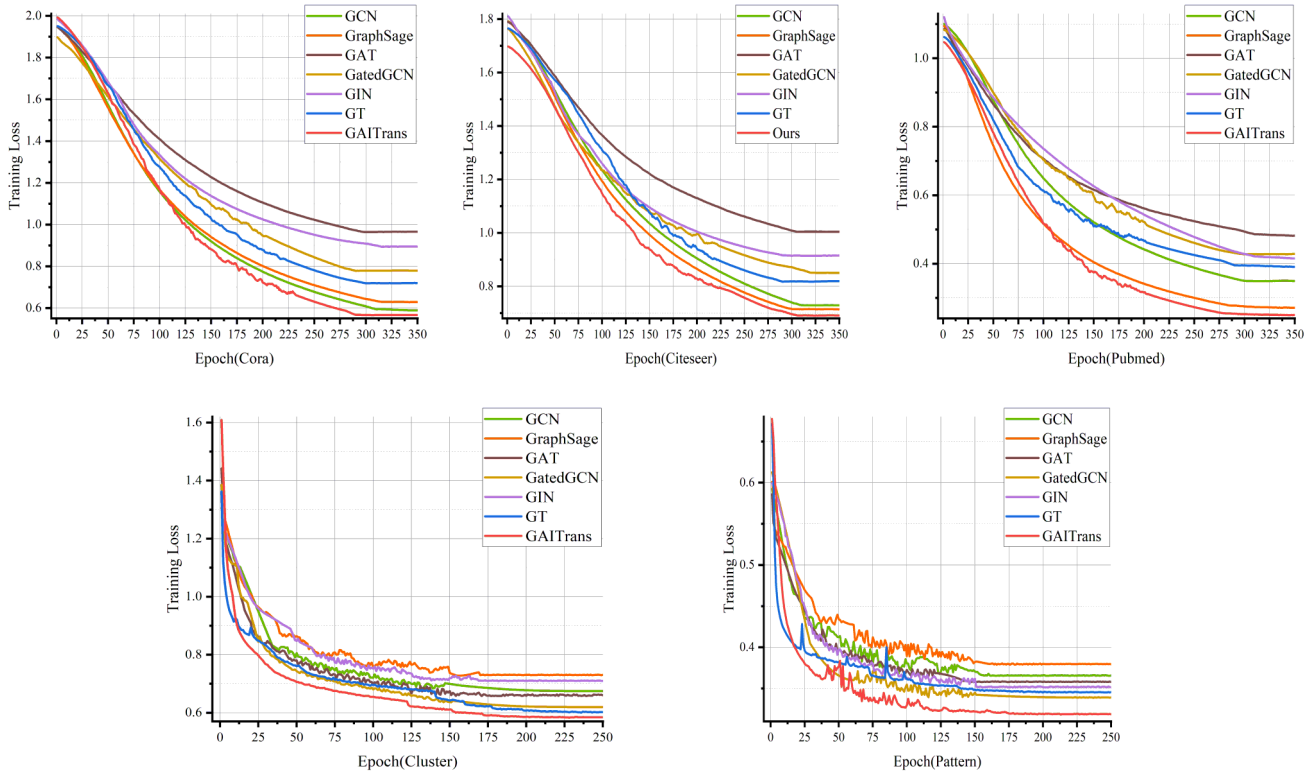


图10 GATrans 与各基线模型的收敛速度及 Loss 损失比较

SMMHA、IRN 组件后,模型各评价指标均具有显著提升,在 3 个不同数据集上最高提升百分比分别为 Cora# {TRNE:2.58%,SMMHA:2.66%,IRN:2.28%}、Cluster# {TRNE:2.52%,SMMHA:2.70%,IRN:2.35%}、Pattern# {TRNE:0.40%,SMMHA:0.73%,IRN:0.42%},这充分验证了本文所提各组件的有效性.此外,通过表中

第 7~10 行数据还发现,不同组件相互组合的性能均优于单独使用某个组件,而且融合所有组件后的模型性能指标比任意两个维度的组件组合更优.因此,上述消融实验结果也进一步验证了本文设计的组件模块对模型节点分类性能均有较好的提升效果.

表 5 各组成模块对模型性能的影响

模型	组成模块				Cora			Cluster			Pattern			
	TRNE	SMMHA			IRN	Acc/%	F_1	Recall/%	Acc/%	F_1	Recall/%	Acc/%	F_1	Recall/%
		GN	GLU	SMM										
M-NONE					80.26	0.804 9	80.31	73.18	0.733 0	73.32	84.67	0.857 2	84.85	
M-TRNE	√				82.84	0.830 2	82.86	75.64	0.758 1	75.84	84.75	0.861 2	84.91	
M-GNGLU		√	√		82.15	0.823 8	82.18	73.78	0.739 5	73.98	84.71	0.860 9	84.86	
M-SMM				√	82.73	0.828 9	82.75	75.52	0.786 9	78.70	84.99	0.863 0	85.12	
M-SMMHA		√	√	√	82.92	0.831 1	82.95	75.83	0.759 9	76.02	85.15	0.864 5	85.36	
M-IRN					√	82.54	0.827 3	82.55	75.46	0.756 5	75.63	0.861 4	84.97	
M-MIX-A	√	√	√	√	83.26	0.835 0	83.31	76.57	0.767 4	76.70	85.27	0.865 4	85.42	
M-MIX-B	√				√	82.95	0.831 3	82.98	75.71	0.758 8	0.7591	84.91	0.862 3	85.10
M-MIX-C		√	√	√	√	83.42	0.836 4	83.47	76.28	0.764 8	76.52	85.84	0.871 5	86.07
GAITrans	√	√	√	√	√	84.25	0.844 5	84.33	77.31	0.775 1	77.62	86.22	0.874 2	86.41

5 结论

由于大多图 Transformer 仅在 Transformer 结构中附加一些辅助模块实现图数据建模,其并未改进模型原有网络结构,建模精度还有待进一步提升.因此,本文提出一种基于图注意力和改进 Transformer 的节点分类方法.该方法首先将压缩图结构信息的节点位置编码融入每层节点输入,进行图结构特征的学习强化.随后,设计二级掩码矩阵并结合 GN、GLU 等优化策略,构造新的多头注意力机制进行节点聚合及更新.最后,以跳跃连接构建 Transformer 层间残差网络,缓解模型退化现象.不同级别数据集上的实验结果一致表明,相较于 6 类基线模型,本文方法在不同评估指标下均可获得最优分类效果,能够全面提升图节点分类任务的处理性能.

参考文献

- [1] 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.
MA S, LIU J W, ZUO X. Survey on graph neural network [J]. Journal of Computer Research and Development, 2022, 59(1): 47-80. (in Chinese)
- [2] 吴博, 梁循, 张树森, 等. 图神经网络前沿进展与应用[J]. 计算机学报, 2022, 45(1): 35-68.
WU B, LIANG X, ZHANG S S, et al. Advances and applications in graph neural network[J]. Chinese Journal of Computers, 2022, 45(1): 35-68. (in Chinese)
- [3] XIE Y, LV S Z, QIAN Y H, et al. Active and semi-supervised graph neural networks for graph classification[J]. IEEE Transactions on Big Data, 2022, 8(4): 920-931.
- [4] BAI L, CUI L X, JIAO Y H, et al. Learning backtrackless aligned-spatial graph convolutional networks for graph classification[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(2): 783-798.
- [5] WU X X, CHENG Q. Stabilizing and enhancing link prediction through deepened graph auto-encoders[C]//Proceedings of the International Joint Conference on Artificial Intelligence. Sydney: IJCAI, 2022: 3587-3593.
- [6] ZHAO T, LIU G, WANG D H, et al. Learning from counterfactual links for link prediction[C]//Proceedings of the International Conference on Machine Learning. New York: PMLR, 2022: 26911-26926.
- [7] SUN Y F, DENG H R, YANG Y, et al. Beyond homophily: Structure-aware path aggregation graph neural network [C]//Proceedings of the International Joint Conference on Artificial Intelligence. Sydney: IJCAI, 2022: 2233-2240.
- [8] CHEN D X, BRAY L O, BORGARDT K M. Structure-aware transformer for graph representation learning[C]//Proceedings of the International Conference on Machine Learning. New York: PMLR, 2022: 3469-3489.
- [9] KIPF T N, WELING M. Semi-supervised classification with graph convolutional networks[C]//Proceedings of the

- International Conference on Learning Representations. Paris: Open Review, 2017: 1-14.
- [10] VELICKOVIC P, CUCURULL G, CASANOVA A, et al. Graph attention networks[C]//Proceedings of the International Conference on Learning Representations. Toronto: Open Review, 2018: 1-12.
- [11] WU F, SOUZA A H, ZHANG T Y, et al. Simplifying graph convolutional networks[C]//Proceedings of the International Conference on Machine Learning. New York: PMLR, 2019: 6861-6871.
- [12] GILMER J, SCHOENHOLZ S S, RILEY P F, et al. Neural message passing for quantum chemistry[C]//Proceedings of the International Conference on Machine Learning. Sydney: PMLR, 2017: 1-10.
- [13] LIU X, SUN D, WEI W. Alleviating the over-smoothing of graph neural computing by a data augmentation strategy with entropy preservation[J]. *Pattern Recognition*, 2022, 132: 108951.
- [14] XU J M, KE H B, CHEN Z W, et al. Over-smoothing relief graph convolutional network-based fault diagnosis method with application to the rectifier of high-speed trains[J]. *IEEE Transactions on Industrial Informatics*, 2022, 19(1): 771-779.
- [15] TOPPING J, GIOVANNI F D, CHAMBERLAIN B P, et al. Understanding over-squashing and bottlenecks on graphs via curvature[C]//Proceedings of the International Conference on Learning Representations. Virtual Conference: Open Review, 2022: 1-30.
- [16] WU Z G, JAIN P, WRIGHT M A, et al. Representing long-range context for graph neural networks with global attention[C]//Proceedings of the Advances in Neural Information Processing Systems. Virtual Conference: NeuralIPS, 2021: 13266-13279.
- [17] HUSSAIN M S, ZAKI M J, SUBRAMANIAN D. Edge-augmented graph transformers: Global self-attention is enough for graphs[C]//Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining. New York: ACM, 2022: 655-665.
- [18] DWIVEDI V P, BRESSON X. A generalization of transformer networks to graphs[C]//Proceedings of the Advances in Neural Information Processing Systems. Virtual Conference: AAAI, 2021: 1-8.
- [19] SUTSKEVER I, VINYALS O, LE Q V. Sequence to sequence learning with neural networks[C]//Proceedings of the Advances in Neural Information Processing Systems. Toronto: NeuralIPS, 2014: 3104-3112.
- [20] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Proceedings of the Advances in Neural Information Processing Systems. New York: NeuralIPS, 2017: 5998-6008.
- [21] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2016: 770-778.
- [22] BA L J, KIROUS J R, HINTON G E. Layer normalization [C]//Proceedings of the Annual Conference on Neural Information Processing Systems. New York: MIT Press, 2016: 1-14.
- [23] MIN E, CHEN R F, XU T Y, et al. Transformer for graphs: An overview from architecture perspective[EB/OL]. (2022-02-17) [2023-06-05]. <http://arxiv.org/abs/2202.08455>.
- [24] YING C Z, CAI T L, LUO S J, et al. Do transformers really perform badly for graph representation?[C]//Proceedings of the Advances in Neural Information Processing Systems. Virtual Conference: NeuralIPS, 2021: 28877-28888.
- [25] DWIVEDI V P, JOSHI C K, LAURENT T, et al. Benchmarking graph neural networks[J]. *Journal of Machine Learning Research*, 2022, 23: 1-48.
- [26] WU Y X, HE K M. Group normalization[C]//Proceedings of the European Conference on Computer Vision. Berlin: Springer, 2018: 3-19.
- [27] DAUPHIN Y N, FAN A, AULI M, et al. Language modeling with gated convolutional networks[C]//Proceedings of the International Conference on Machine Learning. Sydney: PMLR, 2017: 933-941.
- [28] NARANG S, CHUNG H W, TAY Y, et al. Do transformer modifications transfer across implementations and applications?[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Toronto: Association for Computational Linguistics, 2021: 5758-5773.
- [29] XIONG R B, YANG Y C, HE D, et al. On layer normalization in the transformer architecture[C]//Proceedings of the International Conference on Machine Learning. Virtual Conference PMLR, 2020: 10524-10533.
- [30] SEN P, NAMATA G, BILGIC M, et al. Collective classification in network data[J]. *AI Magazine*, 2008, 29(3): 93-93.
- [31] HAMILTON W L, YING R, LESKOVEC J. Inductive representation learning on large graphs[C]//Proceedings of the Annual Conference on Neural Information Processing Systems. New York: Curran Associates, 2017: 1024-1034.
- [32] HUSSAIN M S, ZAKI M J, SUBRAMANIAN D. Residual gated graph convnets[EB/OL]. (2018-04-24) [2023-06-05].

<http://arxiv.org/abs/1711.07553>.

- [33] XU K, HU W, LESKOVEC J, et al. How powerful are graph neural networks?[C]//Proceedings of the International Conference on Learning Representations. New York: Open Review, 2019: 1-17.

作者简介



李 鑫 男,1983年出生,江苏连云港人. 博士,南京邮电大学副教授. 主要研究方向为图机器学习与图挖掘、边缘计算等.
E-mail: xinli@njupt.edu.cn



陆 伟 男,1999年出生,江苏东台人. 现为南京邮电大学在读研究生. 主要研究方向为图机器学习.
E-mail: 1221077007@njupt.edu.cn