

# 基于Graph-Transformer的大规模城市路网 空间结构表示学习方法

白帆, 何炜, 何可佳, 何兵

(成都信息工程大学 计算机学院, 四川 成都 610225)

**摘要:** 路网结构对交通流动效率和安全性具有显著影响。然而, 现有路网数据虽然拓扑结构较为完整, 但辅助信息极为稀疏, 通常难以提取道路网络结构分析所需的特征。因此, 如何有效地通过深度学习挖掘结构中隐含的信息成为交通网络分析重要任务之一。提出一种基于Graph-Transformer的道路网络自监督学习模型, 模型架构由子图采样数据增强、Graph-Transformer编码器和节点-子图对比损失三部分组成。其中, 子图采样旨在提升大规模城市路网分析效率, 自监督对比学习则是用于最大化不同子图视图中正负样本对的互信息, 以捕获和表示道路网络的隐含结构信息, 从而支持下游的道路网络分析任务。实验结果表明, 模型在多个城市数据集上的准确率提升0.02%~2.45%, F1分数提升0.07%~2.55%。同时, 在下游任务社区划分效果上表现出更优的模块度Q值, 提升了0.02%~1.75%。通过研究改进提高了道路网络结构特征提取的有效性, 为分析和理解城市大规模路网隐含结构特征提供了一种新方法。

**关键词:** 图嵌入; 表示学习; 社区检测; 空间结构; 对偶图

DOI: 10.11907/rjtk.242065

中图分类号: TP183

文献标识码: A

文章编号: 1672-7800(2026)004-0065-11

扫描二维码阅读全文:



## Graph-Transformer Based Learning Method for Spatial Structure Representation of Large-scale Urban Road Networks

BAI Fan, HE Wei, HE Kejia, HE Bing

(School of Computer Science, Chengdu University of Information Technology, Chengdu 610225, China)

**Abstract:** The road network structure significantly impacts traffic flow efficiency and safety. However, while existing road network datasets maintain relatively complete topological structures, their auxiliary information is extremely sparse, making it difficult to extract the features required for road network structure analysis. Therefore, effectively mining the implicit information in the structure through deep learning has become one of the important tasks in traffic network analysis. This paper proposes a self-supervised learning model for road networks based on Graph-Transformer. The model architecture consists of three parts: subgraph sampling for data augmentation, a Graph-Transformer encoder, and a node-subgraph contrastive loss. Subgraph sampling is used to improve the efficiency of large-scale urban road network analysis, while self-supervised contrastive learning is employed to maximize the mutual information between positive and negative sample pairs in different subgraph views, thereby capturing and representing the implicit structural information of the road network to support downstream road network analysis tasks. Experimental results demonstrate that the proposed model achieves accuracy improvements of 0.02% to 2.45% and F1 score improvements of 0.07% to 2.55% across multiple city datasets. Additionally, it exhibits better modularity Q values in downstream task community detection and improves by 0.02% to 1.75%. This study enhances the effectiveness of road network structural feature extraction and provides a new method for analyzing and understanding the implicit structural features of large-scale urban road networks.

**Key Words:** graph embedding; representation learning; community detection; spatial structure; dyadic graphs

收稿日期: 2025-03-04

基金项目: 四川省自然科学基金项目(2023NSFSC0243); 成都信息工程大学人才引进计划项目(KYTZ202261)

作者简介: 白帆(1999-), 男, 成都信息工程大学计算机学院硕士研究生, 研究方向为图挖掘与图神经网络; 何炜(2001-), 男, 成都信息工程大学计算机学院硕士研究生, 研究方向为图像处理与深度学习; 何可佳(2000-), 男, 成都信息工程大学计算机学院硕士研究生, 研究方向为低空经济和图神经网络; 何兵(1982-), 男, 博士, 成都信息工程大学计算机学院副教授、硕士生导师, 研究方向为空间信息学与智能计算。本文通讯作者: 何兵。

## 0 引言

路网结构分析是现代城市交通规划与管理的重要研究领域。路网结构不仅影响着交通出行的通畅性,还与社会经济发展、环境保护和公共安全等密切相关。特别是当前城市化进程的加快,道路的扩展和改造不断增加道路网络结构的复杂性<sup>[1]</sup>。因此,如何有效地分析和优化城市路网结构,以提高道路交通系统的效率和安全性,成为学术界和实践领域面临的挑战之一<sup>[2]</sup>。

在复杂的交通网络分析中,研究者们采用了多种复杂网络分析方法,深入探讨了道路网络的拓扑结构。这些方法包括研究网络的无标度特性、小世界效应<sup>[3-4]</sup>、评估网络的结构鲁棒性<sup>[5]</sup>,以及通过度相关性来表征网络的拓扑特征<sup>[6]</sup>。同时,研究者们还运用介数中心性等复杂网络指标来揭示道路网络的增长机制<sup>[7]</sup>及其与交通流量和人类活动的关系<sup>[8-9]</sup>。然而,这些方法存在一定局限性,如介数中心性在预测城市交通流量时可能表现不佳,需要整合其他因素(如距离衰减定律和人类活动的空间异质性)来解释交通流量分布<sup>[10]</sup>。此外,研究表明,道路网络的结构特征对交通流动效率和安全性具有显著影响,如街道布局、交叉口设计等因素<sup>[11-12]</sup>,进一步强调了深入分析和理解复杂城市道路交通网络隐含结构特征的重要性<sup>[13]</sup>。

当前,许多关键的道路网络任务,如旅行时间估计<sup>[14-15]</sup>、交通预测<sup>[16]</sup>以及k最近兴趣点查询<sup>[17-18]</sup>等均依赖于机器学习算法来执行,且通常需要一组特征信息来描述不同的城市道路段。然而,城市道路网络结构除所提供的网络结构外,很少有其他丰富的可用信息,面临复杂的道路网络分析任务时,往往因数据不足而遇到困难。例如,来自众包产生的OSM(OpenStreetMap)数据网络结构较为完整,但辅助信息却非常稀疏,难以派生出执行许多道路网络分析任务所需的特征<sup>[19]</sup>。因此,虽然道路网络结构蕴含丰富的潜在信息,但现有研究表明,精准捕捉并有效挖掘其复杂结构特征并非易事,实际应用效果往往难以达到预期。

近年来,诸如DeepWalk<sup>[20]</sup>、Node2Vec<sup>[21]</sup>、LINE<sup>[22]</sup>以及SDNE<sup>[23]</sup>等典型图嵌入技术被广泛应用于社交网络建模、电子商务产品推荐、图挖掘、自然语言处理和计算机视觉等领域<sup>[24]</sup>。图嵌入算法可以学习网络结构的信息并以节点嵌入的形式存储。这些嵌入是实值向量,具有固定长度,可作为机器学习算法的输入特征,用于节点分类和链接预测等下游任务。例如,Sim2vec通过深度神经网络综合考虑了邻接相似性、可达性相似性和邻域相似性,并将这些节点相似性编码到统一的潜在空间中,以捕获网络结构信息<sup>[25]</sup>。IRN2Vec通过综合考虑交叉点的地理位置、内在属性以及道路使用者的移动行为,进行编码道路网络信息<sup>[26]</sup>。StreetNode2vec通过基于出行路线学习道路网络中

节点的特征表示,进行模型参数校准,以提取交通特征<sup>[27]</sup>。上述方法通常需要依赖行车轨迹等辅助信息,而在实际应用中,这些数据由于数据隐私等原因,难以被大规模应用于路网结构分析中。

随着深度学习的兴起,图神经网络(Graph Neural Network, GNN)也逐步应用到道路网络表示中。现有研究主要通过构建新型图卷积网络来融合结构关系(Road Network Fusion Network, RFN)<sup>[28]</sup>、整合边缘特征<sup>[29]</sup>,以及通过图注意同构网络进行信息聚合(Graph Attention Isomorphic Network, GAIN)<sup>[30]</sup>等方法,来学习道路网络的图形表示。Transformer技术已成为大模型的主要技术架构,也逐步与GNN相结合在图数据上得到了应用。例如,Graphformer是一种将Transformer网络推广到图结构数据的方法,通过引入基于邻域连接的注意力机制、拉普拉斯特征向量位置编码、批归一化和边特征表示,缩小了原始Transformer与图神经网络之间的差距<sup>[31]</sup>。GFormer是一种结合生成式自监督学习与图Transformer架构的推荐系统新方法,通过自动化自监督数据增强和理性化协同发现,提升了用户-物品交互模式的表示学习<sup>[32]</sup>。在这些方法中,NodeFormer通过ALL-Pair消息传递方案和核化Kernelized Gumbel-Softmax算子,实现了在大型图上高效学习潜在图结构和节点分类的能力<sup>[33]</sup>。

尽管,经典图表示学习、GNN和Transformer技术已在社交网络图数据上得到了大量应用及验证,但社交网络通常是强连接的,而道路网络却常常面临空间表示不准确的问题,甚至因“岛屿”的存在而导致路段连接被断开。因此,道路网络的空间结构表示学习与社交网络等其他类型的网络,在结构和连通性上有显著差异。部分研究致力于将图嵌入技术引入道路网络,然而,仅有少数研究尝试将这些图嵌入直接应用于下游任务,如速度限制预测和到达时间估计<sup>[34-36]</sup>。这些方法在评估它们捕获结构信息的能力方面存在不足,同时也没有采用基于Transformer和GNN的方法来提取道路网络上的行驶语义<sup>[37]</sup>。综上,上述方法难以适应大规模静态图的结构特征学习,如何有效地利用这些技术来捕捉和表示道路网络的复杂拓扑结构,以便更好地支持机器学习任务,仍是一个开放性问题。本文提出了一种针对大规模城市路网结构数据分析的自监督学习模型,通过子图采样的方式降低模型的显存消耗,借助Transformer提取节点及结构的潜在特征,以子图正负样本对比的形式,实现网络训练和特征提取,并在下游任务社区检测上验证所提取特征的有效性。

## 1 方法

### 1.1 问题及定义

给定一个图 $G = (X, A)$ ,其中,包含一组节点特征 $X = \{x_1, x_2, \dots, x_N\}$ ,图中的节点数为 $N$ , $x_i \in \mathbb{R}^f$ 表示节点 $i$ 有 $f$

维的特征。节点间的关系信息,以邻接矩阵的形式表示,  $A \in \mathbb{R}^{N \times N}$ 。图是未加权的,如果图中存在一条边,邻接矩阵中的相应元素为 1,即  $A(i,j) = 1$ ; 否则为 0,即  $A(i,j) = 0$ 。由于大规模路网结构的邻接矩阵处理所需的显存占用较大,现有模型难以高效处理,本文提出一种结合子图采样的对比自监督学习方法,用以捕获结构特征信息。在采样方法中,每个中心节点  $i$  通过子图采样器  $S$  从原始图中提取其上下文子图  $X_i \in \mathbb{R}^{N' \times F}$ 。其中,  $X_i$  表示为第  $i$  个节点的上下文子图特征,  $A_i$  表示节点  $i$  及其邻居节点之间的关系信息,  $N'$  为上下文子图的大小。本文对比监督学习模型是学习一个编码器  $E: \mathbb{R}^{N' \times F} \times \mathbb{R}^{N' \times N'}$  到节点  $i$  的区域结构信息,从而为子图中的每个节点生成表示。

### 1.2 模型框架

城市路网结构自监督对比学习模型结构如图 1 所示。

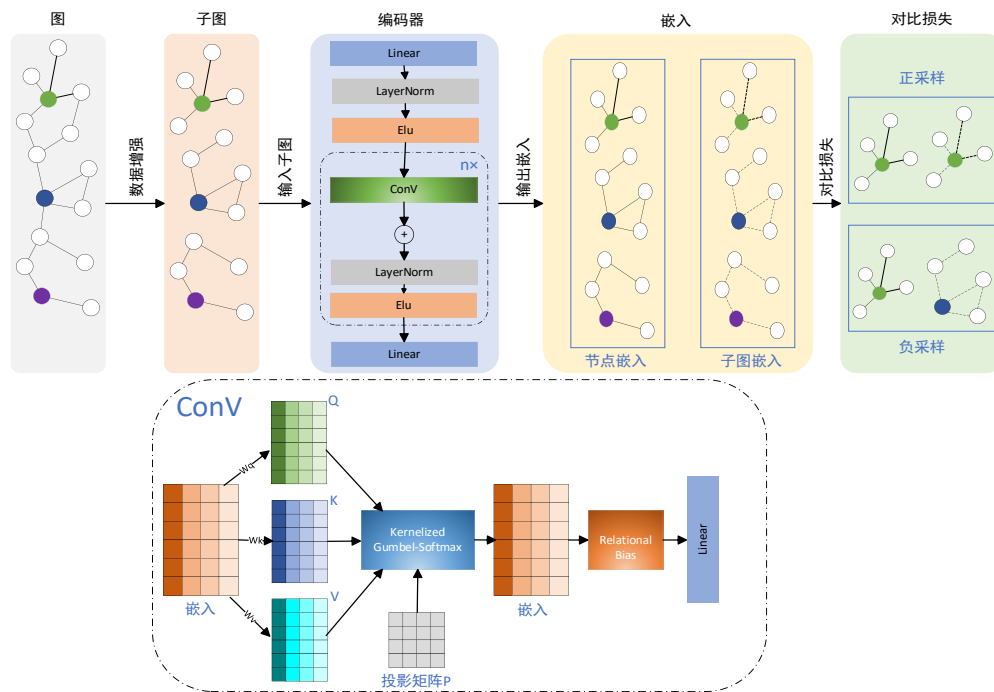


Fig. 1 Structure diagram of the learning model for urban road network structural features

图 1 城市路网结构特征学习模型结构图

### 1.3 子图采样数据增强

鉴于路网数据中节点与其局部邻域之间的关系通常比与远距离节点的关系更加紧密,而大规模图的处理又常常受限于显存容量,本文采用 PageRank 算法从原始图中采样一系列包含局部邻域信息的子图,以提高图结构学习任务中的样本质量和有效性,更好地捕获城市路网的复杂结构关系,适应更多节点的大型路网数据<sup>[38]</sup>。PageRank 算法是一种基于图结构的随机游走方法,通过计算每个节点的相对重要性得分,评估节点在整个图中的“中心性”或“影响力”。此外,重要性高的节点被优先采样,从而确保选出的子图更加精确地反映目标节点的局部结构。重要性得分矩阵  $S$ , 如公式(1)所示:

$$S = \alpha \cdot (I - (1 - \alpha) \cdot \bar{A}) \quad (1)$$

其中,  $I$  是单位矩阵,  $\alpha$  是重启随机游走概率,通常设置为 0.15<sup>[39]</sup>。  $\bar{A}$  表示列标准化的邻接矩阵。

$S(i, :)$  是节点  $i$  的重要性得分向量,反映与其他节点的相关性。对于特定节点  $i$ , 子图采样器  $S$  选择前  $k$  个重要邻居来构成子图。得分矩阵  $S$  中的索引,如公式(2)所示:

$$idx = top\_rank(S(i, :), k) \quad (2)$$

其中,  $top\_rank$  是返回前  $k$  个值的索引函数,  $k$  表示上下文子图的大小。子图采样器  $S$  将使用节点索引处理原始图,以获得节点  $i$  的上下文子图  $\mathcal{G}_i$ 。它的邻接矩阵  $X_i$  和特征矩阵  $A_i$ , 如公式(3)所示:

$$X_i = X_{idx, :}, A_i = A_{idx, idx} \quad (3)$$

在模型中,首先输入一个图,接着利用 PageRank 算法计算节点重要性得分,通过优先选取前  $k$  个高分邻居,并提取子图的邻接矩阵和特征矩阵,生成能包含保留目标节点的局部邻域信息与上下文信息的子图,再进行数据增强,得到特定节点对应的子图。然后,将这些子图输入到编码器进行处理。编码器以中心节点的上下文子图为输入,通过线性变换、层归一化和 ELU 激活等操作获取节点和子图嵌入。为降低计算复杂度,编码器采用稀疏注意力机制,并结合 Kernelized Gumbel-Softmax 实现结构稀疏化,同时,随机特征映射与内积核函数聚合特征实现嵌入的更新。最后,模型通过构造正负样本对,利用对比学习优化正负样本的对比损失,以学习图结构表示。损失函数采用破坏函数生成负样本,并利用边际三重损失缩小正样本距离和扩大负样本距离,从而提升节点表示质量,进行节点—子图对比损失训练。

其中,  $idx$  是索引操作,  $X_i$  是按行(即节点)索引的特征矩阵。  $A_{idx,idx}$  是对应于子图的按行和列索引的邻接矩阵。

#### 1.4 编码器

给定中心节点的上下文子图  $G_i = (X_i, A_i)$ , 用编码器  $E: \mathbb{R}^{N' \times F} \times \mathbb{R}^{N' \times N'} \rightarrow \mathbb{R}^{N' \times F'}$  对其进行编码来获得潜在表示矩阵  $H_i$ , 如公式(4)所示:

$$H_i = E(X_i, A_i) \quad (4)$$

Graph-Transformer 编码器  $E$  采用 NodeFormerConv 作为核心架构, 包括多个图卷积层。在卷积层的 ConV 模块中, 通过融合多头注意力机制, 生成投影矩阵、Kernelized Gumbel-Softmax 注意力计算和图关系偏置引入, 从而进行节点表示学习。每一个图卷积层都实现了节点的聚合更新和注意力权重计算。

##### 1.4.1 投影矩阵

投影矩阵用于将高维数据投影到低维空间中, 这样可以有效减少数据的维度, 保留数据的主要特征。首先, 计算完整的  $d \times d$  的块  $n_{blocks}$ , 如公式(5)所示:

$$n_{blocks} = \left\lfloor \frac{m}{d} \right\rfloor \quad (5)$$

其中,  $\lfloor \cdot \rfloor$  表示取整计算。

每个完整的块(从1到  $n_{blocks}$ ) 生成一个随机矩阵, 并通过  $QR$  分解得到正交矩阵, 如公式(6)所示:

$$U_i \sim N(0, 1)^{d \times d}, Q_i = (QR(U_i))^T \quad (6)$$

其中,  $U_i$  是标准正态分布的随机矩阵,  $Q_i$  为正交矩阵。每个  $Q_i$  将被添加到一个列表中。

当矩阵的行数  $m$  不能被  $d$  整除, 对存在的剩余行生成新的随机矩阵, 并通过  $QR$  分解得到正交矩阵并保留前  $r$  行, 如公式(7)所示:

$$U_{rem} \sim N(0, 1)^{d \times d}, Q_{rem} = (QR(U_{rem}))^T, Q_{rem} = Q_{rem}[0:r] \quad (7)$$

随后, 将正交矩阵  $Q_i$  和  $Q_{rem}$  进行拼接, 得到矩阵  $Q$ , 如公式(8)所示:

$$Q = \begin{bmatrix} Q_1 \\ Q_2 \\ \vdots \\ Q_{n_{blocks}} \\ Q_{rem} \end{bmatrix} \quad (8)$$

该矩阵的大小为  $m \times d$ , 其中,  $m$  是行数,  $d$  是列数。

最后, 对矩阵  $Q$  进行随机缩放, 根据每一行的范数生成一个随机缩放因子  $multiplier_i$  组成的矩阵  $diag(multiplier_i)$ , 并乘以矩阵  $Q$  得到投影矩阵  $p$ , 如公式(9)–(10)所示:

$$multiplier_i = \|V_i\|, V_i \sim N(0, 1)^d \quad (9)$$

$$p = diag(multiplier_i) \cdot Q \quad (10)$$

其中,  $V_i$  是标准正态随机向量,  $\|V_i\|$  为范数,  $diag(multiplier_i)$  为对角矩阵, 对角线元素为  $multiplier_i$ 。

##### 1.4.2 Kernelized Gumbel-Softmax 注意力计算

Kernelized Gumbel-Softmax 是一种高效的注意力聚合

方法, 它利用 Gumbel Softmax 技术对查询、键和值进行核化转换, 以降低注意力计算的复杂度, 实现所有对节点之间的注意力快速聚合。通过 query、key 和 value 三者的线性变换, 构造多头注意力机制, 然后对 query 和 key 进行温度缩放和核化转换, 温度用于控制 Gumbel-Softmax 采样的“硬度”, 核化转换使用投影矩阵对 query 和 key 进行映射, 最后计算 Gumbel 加权聚合和边的注意力权重。

在计算注意力的过程中, 为避免梯度消失, 采用 Gumbel 分布进行采样来模拟离散化过程。Gumbel 采样公式如公式(11)所示:

$$G = -\log(-\log(U)), U \sim Uniform(0, 1) \quad (11)$$

随后, 将 Gumbel 噪声应用到 key 上, 得到公式(12):

$$K'_i = \hat{K} \times \exp(G) \quad (12)$$

其中,  $\hat{K}$  是 key 的变换结果,  $\exp(G)$  是对 Gumbel 噪声的指数运算。

计算 Gumbel 加权聚合, 使用 Gumbel 采样的 key 和 query 计算加权的聚合值。计算分子  $z_{num}$  和分母  $z_{den}$ , 如公式(13)所示:

$$\begin{aligned} z_{num} &= \text{numerator\_gumbel}(\hat{Q}', K_i, V) \\ z_{den} &= \text{denominator\_gumbel}(\hat{Q}', K_i) \end{aligned} \quad (13)$$

其中,  $\text{numerator\_gumbel}$  是一个用于计算加权值的函数,  $V$  是节点的特征,  $\text{denominator\_gumbel}$  计算归一化因子, 用于标准化加权聚合结果。

更新后的节点嵌入  $z_{output}$ , 如公式(14)所示:

$$z_{output} = \frac{1}{K} \sum_{i=1}^K \frac{z_{num}}{z_{den}} \quad (14)$$

其中,  $k$  是 Gumbel 采样的次数, 最终通过对  $k$  次采样的结果求平均, 得到节点的更新嵌入。

计算边的注意力权重, 使用每条边的 query 和 key 的点积除以边的归一化因子, 从而得到每条边的最终注意力权重。边的注意力权重通常用于表示节点间的关系强度。Gumbel-Softmax 允许在保持可微性的同时模拟离散采样, 避免了在训练时离散采样带来的不可导问题, 从而使得模型可以通过梯度下降进行优化。

##### 1.4.3 图关系偏置引入

选择引入图的关系性偏置(Relational Bias), 进一步改进邻居信息的聚合, 以捕捉输入数据中的复杂关系。目标是计算图中节点的表示, 并在计算过程中考虑输入图邻接矩阵的关系偏置。

首先, 对每个节点的入度和出度度数计算并标准化, 每个偏置值  $b[i]$  应用 Sigmoid 激活函数, 如公式(15)所示:

$$b_i = \sigma(b[i]) = \frac{1}{1 + \exp(-b[i])} \quad (15)$$

然后, 构造稀疏加权邻接矩阵, 每条边的权重是由  $b_i$  和标准化度数因子  $d_{norm\_in}$ 、 $d_{norm\_out}$  确定的。对于每条边, 其权重  $value$  计算, 如公式(16)所示:

$$value = b_i \cdot d_{norm\_in}[col] \cdot d_{norm\_out}[row] \quad (16)$$

得到的稀疏邻接矩阵表示节点之间的关系。接下来对矩阵进行卷积操作,对于每个特征维度*i*,使用加权邻接矩阵 $adj_i$ 和节点特征 $x[:, :, i]$ 进行卷积操作,得到更新后的节点表示 $conv\_output_i$ ,如公式(17)所示:

$$conv\_output_i = matmul(adj_i, x[:, :, i]) \quad (17)$$

其中, $matmul$ 表示矩阵乘法操作, $adj_i$ 是加权邻接矩阵, $x[:, :, i]$ 是第*i*个特征维度的节点特征。最后,将各个通道的卷积结果堆叠在一起,形成最终的输出。通过对每个边引入归一化的关系偏置,模型能够更加灵活地捕捉图中节点之间的相互关系,从而提高模型在图结构数据上的表现。

### 1.5 节点—子图对比损失

本文采用自监督对比学习,通过优化正负样本对比损失,学习图结构的表示。对于每个节点,上下文子图中局部信息的节点表示 $h_i$ ,将其对应的上下文子图表示 $s_i$ 视为正样本。同时,通过一个破坏函数 $P$ 来生成负样本,如公式(18)所示:

$$\{\tilde{s}_1, \tilde{s}_2, \dots, \tilde{s}_M\} \sim P(\{s_1, s_2, \dots, s_m\}) \quad (18)$$

其中, $m$ 表示子图集合的大小。

采用边际三重损失来优化模型,以便有效区分正负样本,从而获得高质量的节点表示。边际三重损失通过对正样本和负样本之间的距离,使正样本尽可能接近,同时使负样本尽可能远离。这种损失函数能够帮助模型在学习过程中更好地捕捉节点的局部结构信息,从而提高表示的质量和区分度。损失表示如公式(19)所示:

$$\mathcal{L} = \frac{1}{M} \sum_{i=1}^M \mathbb{E}_{(x, \lambda)} \left( -\max(\sigma(h_i s_i) - \sigma(h_i \tilde{s}_i) + \epsilon, 0) \right) \quad (19)$$

其中, $\sigma(x) = 1/(1 + \exp(-x))$ 是sigmoid函数。

## 2 实验

### 2.1 评价标准

本文采用准确率(Accuracy, ACC)、模块度(Modularity, Q)和F1分数来评估模型的实验结果。

#### 2.1.1 准确率

准确率(Accuracy, ACC)衡量了模型正确预测的样本数占总样本数的比例。准确率越高,表示模型的预测越准确。对于二分类问题,准确率的计算公式如公式(20)所示:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (20)$$

其中, $TP$ 是模型正确预测为正类的样本数, $TN$ 是模型正确预测为负类的样本数, $FP$ 是模型错误预测为正类的负类样本数, $FN$ 是模型错误预测为负类的正类样本数。

#### 2.1.2 模块度

模块度(Modularity, Q)是衡量图划分的质量,用于评估模型在图结构中的可视化效果<sup>[40]</sup>。模块度是衡量图中节点被正确划分到各个社区的程度,模块度越高,表示划

分越合理。模块度的计算公式如公式(21)所示:

$$Q = \frac{1}{2m} \sum_{i,j} \left( A_{ij} - \frac{k_i k_j}{2m} \right) \quad (21)$$

其中, $A_{ij}$ 表示图中节点*i*和节点*j*之间的连接关系, $k_i$ 和 $k_j$ 分别是节点*i*和节点*j*的度数, $m$ 是图中边的总数, $c_i$ 和 $c_j$ 分别是节点*i*和节点*j*所在的社区。当节点*i*和节点*j*属于同一社区时, $\delta(c_i, c_j)$ 为1,否则为0。

#### 2.1.3 F1分数

路网数据中正负样本的比例差异较大时,F1分数可综合评估模型性能并反映模型的实际表现。它是精确率(Precision)和召回率(Recall)的调和平均值,能够平衡模型在正类预测上的准确性和完整性。

精确率和召回率的计算公式如公式(22)–(23)所示:

$$Precision = \frac{TP}{TP + FP} \quad (22)$$

$$Recall = \frac{TP}{TP + FN} \quad (23)$$

其中, $TP$ 表示正确识别的正样本数, $FP$ 表示错误识别为正样本的负样本数, $FN$ 表示错误识别为负样本的正样本数。

F1分数的计算公式如公式(24)所示:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (24)$$

其中,F1分数的取值范围为0到1,数值越大表明模型的精确率和召回率越平衡,模型性能越好。

## 2.2 实验数据

道路网络数据来源于哈佛大学数据平台(Harvard Dataverse)的全球城市街道网络项目(Global Urban Street Networks)<sup>[41]</sup>。寇世浩等<sup>[42]</sup>通过聚类将中国城市路网分为4类,本文从每一类中选取了一些代表性城市,包括成都、武汉、郑州、贵阳、天津、太原、青岛、重庆和长沙等9个城市,并为此构建了数据集。该数据集分别包含了66 326、38 070、19 172、8 019、17 271、16 918、36 799、24 264和30 511条路段,如表1所示。

Table 1 Basic information of different datasets

表1 不同数据集的基本情况

数据集	节点	路段	面积/Km <sup>2</sup>
成都	28 216	66 326	3 347
武汉	16 853	38 070	1 604
郑州	9 249	19 172	1 032
贵阳	4 134	8 019	789
天津	8 466	17 271	1 260
太原	8 231	16 918	855
青岛	14 509	36 799	1 511
重庆	11 713	24 264	1 266
长沙	13 287	30 511	1 403

由于路网数据中的节点辅助信息有限,同时,为了验证图嵌入模型对道路网络结构信息捕获的有效性,本文采用对偶图的方法来以节点为主的拓扑结构转换为以边为主的对偶图,如图2所示。首先,初始化对偶图并构建原

图的边与对偶图节点的映射;其次,对每个对偶图节点(即原始图的边)添加长度、车道数和等级等属性信息作为对偶图的节点特征;其次,为便于后续的图对比学习,提取对偶图中节点的车道数作为标签,并计算得到对偶图的邻接矩阵以支持图神经网络的信息传递;最后,在实验过程中,对每个城市的图结构数据,按照需求生成相应的用于训练、验证和测试的数据集。

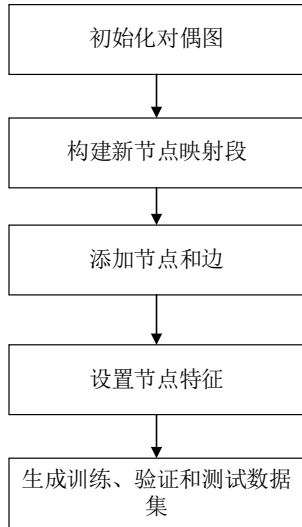


Fig. 2 Dual graph processing flow

图2 对偶图处理流程

### 2.3 实验环境

实验环境采用Linux操作系统,以Python 3.9、PyTorch 2.3.1和PyCharm作为开发环境。具体实验配置信息如表2所示。

Table 2 Experimental environment configuration

表2 实验环境配置

硬件配置	参数
CPU	Intel Xeon Platinum 8255C CPU @ 2.50GHz
GPU	NVIDIA GeForce RTX 4070S
内存	32 G
磁盘	16 G
显存	16 G
CPU 核心数	8

### 2.4 实验结果

为验证本文所提模型的有效性,选择以BGRL<sup>[43]</sup>、Subg-Con<sup>[39]</sup>、GCN(Graph Convolutional Networks)<sup>[44]</sup>、GraphSAGE<sup>[45]</sup>、DeepWalk<sup>[20]</sup>和Node2vec<sup>[21]</sup>作为基准模型。其中,BGRL是一种自监督图表示方法,通过对比学习消除对负例的依赖,使用2个图编码器学习节点表示;Subg-Con基于子图对比的自监督学习,利用中心节点与子图的关联性提升节点表示;GCN是图卷积神经网络,适用于图结构数据的特征聚合和节点关系学习;GraphSAGE是一个动态图数据的节点表示框架,通过聚合邻居信息更新节点特征;DeepWalk通过随机游走路径序列化节点,并用skip-gram模型学习节点嵌入;Node2vec通过引入有偏随机游走

策略,能够探索节点的局部邻域和全局结构,并利用Skip-Gram模型学习节点的低维嵌入。

#### 2.4.1 定量结果

在本文的实验中,对比了图神经网络模型(BGRL、Subg-Con、GCN、GraphSAGE、DeepWalk、Node2vec以及论文所提模型)在9个不同城市(成都、武汉、郑州、贵阳、天津、太原、青岛、重庆和长沙)数据集上的准确率(ACC)的表现,如表3所示。

Table 3 Comparison of accuracy for each model on different datasets

表3 不同数据集下各模型的准确率对比

方法	成都	武汉	郑州	贵阳	天津	太原	青岛	重庆	长沙
BGRL	87.43	87.02	96.64	76.99	89.38	80.64	92.45	80.69	84.60
Subg-Con	<b>87.48</b>	<b>87.18</b>	<b>97.27</b>	<b>77.24</b>	<b>90.23</b>	<b>80.68</b>	<b>92.81</b>	<b>81.15</b>	<b>84.86</b>
GCN	87.40	87.15	96.73	76.23	89.35	80.33	92.38	80.24	84.26
GraphSAGE	87.30	84.90	96.82	77.22	89.38	80.33	92.04	79.85	84.75
DeepWalk	87.37	85.94	97.12	76.91	89.19	80.37	92.08	80.44	84.20
Node2vec	87.35	86.02	96.41	77.13	89.19	80.05	92.05	80.55	84.87
Ours	<b>88.10</b>	<b>87.35</b>	<b>97.29</b>	<b>77.37</b>	<b>90.25</b>	<b>80.75</b>	<b>92.83</b>	<b>81.28</b>	<b>85.21</b>

由表3可知,所提模型在多数城市数据集上的准确率有显著提升。具体而言,本模型在多个城市数据集上的性能均优于次优模型,具体提升幅度如下:成都(+0.62%)、武汉(+0.17%)、郑州(+0.02%)、贵阳(+0.13%)、天津(+0.02%)、太原(+0.07%)、青岛(+0.02%)、重庆(+0.13%)和长沙(+0.35%)。通过对比各模型在不同城市数据集上的表现,可以发现本模型在整体准确率上占据优势。相较于传统图嵌入模型(GraphSAGE、DeepWalk和Node2vec),Subg-Con和本模型在大多数数据集上相较传统模型更具竞争力。

表4展示了本文所提模型与其他6个模型在9个数据集上F1分数的表现情况。

Table 4 Comparison of F1-score for each model on different datasets

表4 不同数据集下各模型的F1分数对比

方法	成都	武汉	郑州	贵阳	天津	太原	青岛	重庆	长沙
BGRL	82.67	80.38	94.98	64.28	83.57	72.13	89.34	71.36	78.57
Subg-Con	<b>83.20</b>	<b>81.37</b>	<b>96.07</b>	<b>69.07</b>	<b>86.00</b>	<b>72.95</b>	<b>89.39</b>	<b>72.10</b>	<b>78.83</b>
GCN	82.30	81.17	95.39	67.59	85.94	71.57	89.26	71.48	78.48
GraphSAGE	82.12	80.42	95.43	68.22	85.38	72.58	88.93	71.08	78.78
DeepWalk	82.34	80.69	96.13	67.24	84.78	72.60	89.14	71.02	78.60
Node2vec	82.50	80.45	95.49	69.01	84.80	72.45	89.32	72.03	78.58
Ours	<b>83.49</b>	<b>81.52</b>	<b>96.28</b>	<b>69.36</b>	<b>86.12</b>	<b>73.84</b>	<b>89.46</b>	<b>72.24</b>	<b>79.08</b>

由表4可知,本模型在多数城市数据集上的F1分数有显著提升。具体而言,本模型在多个城市数据集上的性能均显著优于次优模型,具体表现为:成都(+0.29%)、武汉(+0.15%)、郑州(+0.21%)、贵阳(+0.29%)、天津(+0.12%)、太原(+0.89%)、青岛(+0.07%)、重庆(+0.14%)和长沙(+0.25%)。其中,太原数据集的性能提升最为显著,达到0.89个百分点。此外,本模型在其他数据集上的表现也较为稳定,如郑州数据集F1分数达到96.28%,表明模型在处理具有规律性空间结构城市路网中的高效性。然而,尽管贵阳数据集规模较小,但本模型仍能取得较高的F1分数,

表明其在不同规模和复杂度的城市路网中具有较强的泛化性和鲁棒性。

表 5 展示了在 9 个路网数据集中,应用不同的模型计算,从而得到的模块度 Q 值。实验结果可以看出,在所选取的所有城市数据集中,本模型的模块度 Q 值普遍较高,表现出更优的社区划分效果。其中,Subg-Con 方法在武汉、贵阳、天津、太原、青岛和长沙 6 个数据集取得了次优,显示出较强的社区划分能力;BGRL 在成都、郑州和重庆 3 个数据集上取得了次优;GCN、GraphSAGE、DeepWalk 和 Node2vec 方法的模块度 Q 值普遍较低,表明其在社区划分上的效果略逊于其他方法。

2.4.2 可视化结果

图 3 选取了 6 个不同的模型,对 6 个城市路网数据集进行了可视化,并进一步采用社区检测算法对路网数据的边嵌入进行了分区分析。本文研究探讨了不同模型在不同城市路网数据集上的表现,并分析了分区效果与路网结构特征之间的关系。从整体视觉效果来看,本文模型可视化结果清晰,整体布局结构合理,便于理解网络的空间结

Table 5 Modularity of different models after partitioning on nine datasets

表 5 不同模型在 9 个数据集上分区后的模块度

方法	成都	武汉	郑州	贵阳	天津	太原	青岛	重庆	长沙
BGRL	<b>96.88</b>	96.74	<b>94.60</b>	93.62	96.05	94.94	96.52	<b>96.51</b>	96.40
Subg-Con	96.57	<b>97.20</b>	94.47	<b>94.15</b>	<b>96.75</b>	<b>95.68</b>	<b>96.70</b>	96.42	<b>97.16</b>
GCN	96.61	96.11	94.29	92.78	95.42	93.94	95.91	95.35	95.96
GraphSAGE	96.80	96.07	93.81	93.94	95.60	94.61	96.45	95.92	96.18
DeepWalk	96.76	97.11	94.02	94.05	96.24	95.33	96.48	95.77	96.59
Node2vec	96.55	97.18	94.25	94.17	96.72	95.59	96.67	95.71	96.78
Ours	<b>96.97</b>	<b>97.24</b>	<b>94.86</b>	<b>94.19</b>	<b>96.88</b>	<b>95.69</b>	<b>96.73</b>	<b>96.56</b>	<b>97.17</b>

构。同时,模型清晰地揭示了城市路网的分区结构,如中心城区与外围区域的划分。

图 4 展示了本文模型在 6 个数据集上的可视化图。由图可知,太原与贵阳的可视化结果(小于贵阳的实际城区)能够较好地分辨 2 个城市的空间结构。太原市的核心城区与郊区之间存在明显的区块结构,这种结构上的差异可能与城市的地理环境、城市扩展模式和行政区划密切相关。天津与郑州的可视化结果表现出了明显的南北分区

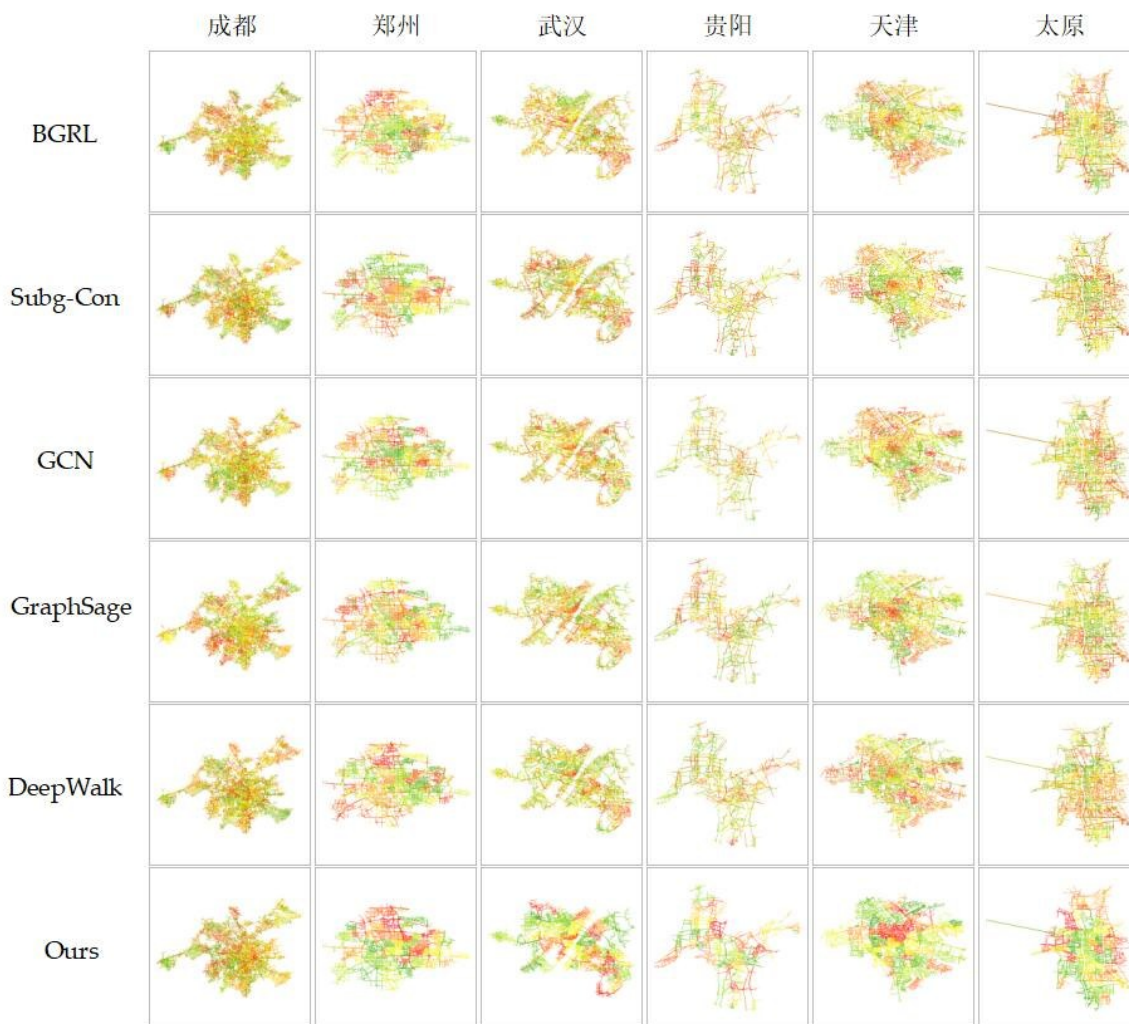


Fig. 3 Visualization of road network partitions of different models

图 3 不同模型的路网分区可视化结果

结构。天津南部的河西、和平与河东区块则表现出更高的密度,表明经济活动和城市发展较为紧密。郑州市的城市结构在北边呈现出明显的区块结构,西南区域则更加密集,黄河的存在可能很大程度上影响了郑州的城市扩展方向。武汉市的汉口区作为武汉的商业和贸易中心,区块结构密集,而武昌区地形起伏较大,自然条件导致其区块特征相对破碎。成都市由于人口稠密且城市发展需求巨大,呈现出较为杂乱的特点,能够反映出城市在空间布局上的复杂性和多样性。

### 2.4.3 模型效率结果分析

表6为本模型与BGRL和Subg-Con在6个数据集上的训练时间和内存消耗。从表6可知,本模型在各个数据集上的训练时间表现尤为突出,在除贵阳数据集外的所有数据集中均实现了最短的训练时间。在成都数据集上,模型的训练时间较BGRL和Subg-Con分别缩短了20.8%和24.8%。在武汉数据集上,模型是三者中最短,较BGRL缩短了15.8%。整体来看,随着数据量的增加训练时间将有所提升。在内存消耗方面,本文模型也普遍低于BGRL和Subg-Con。特别是在成都、武汉、天津3个数据集上,本文模型的内存使用量大大低于BGRL。在成都数据集上,本文模型的内存消耗相较BGRL减少了82.7%、Subg-Con减少了63.9%。尽管在贵阳数据集上,本文模型的训练时间比Subg-Con略低,但考虑到内存消耗的优势以及在其他数据集上的综合表现,本文模型在训练效率和资源利用上的平衡表现使其在实际应用中具有很大的潜力。

## 2.5 消融实验

### 2.5.1 不同损失函数实验结果对比

在本次消融实验中,本文针对图神经网络中的节点表

Table 6 Training time and memory size for each model

表6 各个模型的训练时间和内存大小

数据集	方法	训练时间	内存
成都	BGRL	2.64S	5080MB
	Subg-Con	2.78S	2440MB
	Ours	2.09S	880MB
武汉	BGRL	1.46S	2968MB
	Subg-Con	1.45S	1882MB
	Ours	1.23S	878MB
郑州	BGRL	0.83S	1848MB
	Subg-Con	0.67S	1290MB
	Ours	0.58S	876MB
贵阳	BGRL	0.30S	912MB
	Subg-Con	0.20S	986MB
	Ours	0.29S	826MB
天津	BGRL	1.23S	2794MB
	Subg-Con	1.09S	2472MB
	Ours	0.93S	878MB
太原	BGRL	0.71S	1504MB
	Subg-Con	0.62S	1490MB
	Ours	0.55S	828MB

示学习任务,对比分析了3种不同的目标函数:Margin损失、Logistic损失以及贝叶斯个性化排名损失(Bayesian Personalized Ranking, BPR)。为全面评估目标函数的效果,本文在9个城市数据集上进行了实验,实验结果如表7所示。

Table 7 Comparison of results of different loss functions

表7 不同损失函数结果对比

	成都	武汉	郑州	贵阳	天津	太原	青岛	重庆	长沙
Margin	<b>88.10</b>	<b>87.35</b>	<b>97.29</b>	<b>77.37</b>	<b>90.25</b>	<b>80.75</b>	<b>92.83</b>	<b>81.28</b>	<b>85.21</b>
Logistic	87.42	<b>87.31</b>	97.28	77.35	90.23	80.27	92.45	<b>80.82</b>	85.12
BPR	<b>87.53</b>	87.29	<b>97.29</b>	<b>77.35</b>	<b>90.24</b>	<b>80.39</b>	<b>92.75</b>	80.78	<b>85.15</b>

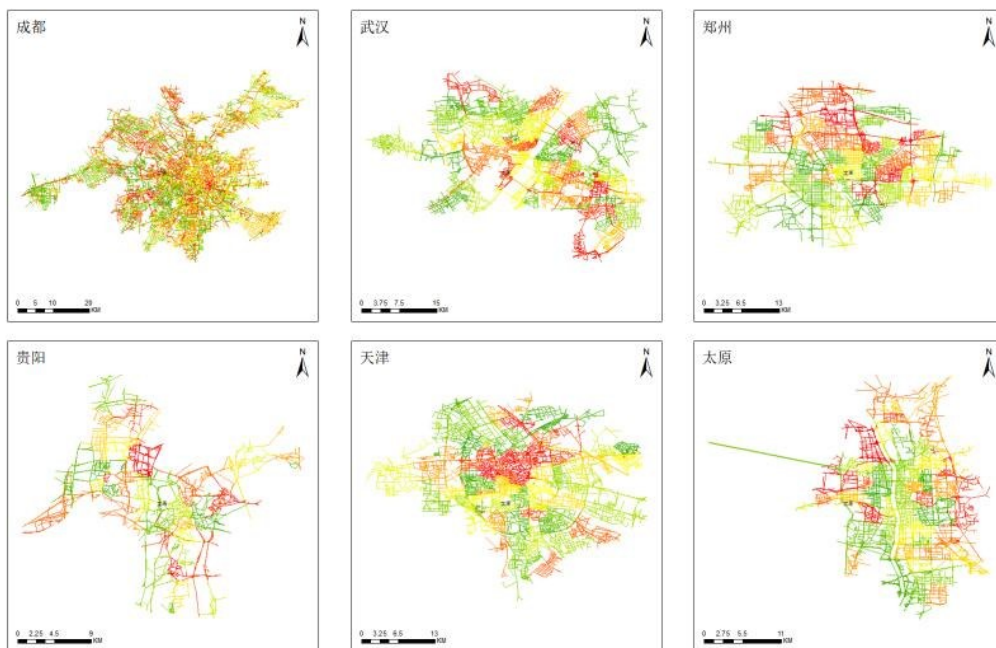


Fig. 4 Visualization of road network partitions of the proposed model

图4 本模型的路网分区可视化结果

由表 7 可知,Margin 损失函数在 9 个城市数据集上的表现均优于 Logistic 损失函数和 BPR 损失函数。其中,Margin 损失函数在成都数据集上的准确率达到 88.10%,比 Logistic 损失函数高出 0.68 个百分点,比 BPR 损失函数高出 0.57 个百分点;在武汉数据集上,Margin 损失函数的准确率为 87.35%,同样高于其他 2 种损失函数。这一趋势在其他城市数据集上也得到了验证,表明 Margin 损失函数在节点表示学习任务中具有优越性。

2.5.2 不同编码器结果对比

本文选取了 3 种主流的图神经网络编码器进行对比,包括图卷积网络(GCN)、图注意力网络(Graph Attention Networks, GAT)以及 GraphSAGE。每个编码器在 9 个不同的数据集上的准确率如表 8 所示。

Table 8 Experimental results of different encoders

表 8 不同编码器下的实验结果

编码器	成都	武汉	郑州	贵阳	天津	太原	青岛	重庆	长沙
GCN	87.48	87.18	97.27	77.24	90.23	80.68	92.81	81.15	84.86
GAT	87.02	87.24	97.17	77.24	90.21	80.27	92.45	81.20	84.75
GraphSAGE	87.12	87.16	97.22	76.86	89.38	80.04	92.23	80.85	84.72
Ours(NodeFormerConv)	88.10	87.24	97.29	77.37	90.25	80.75	92.83	81.28	85.21

本文采用 NodeFormerConv 作为编码器。从表 8 中可以看出,NodeFormerConv 作为图神经网络编码器在各数据集上均展现出了优于或至少与其他编码器持平的性能,这表明 Graph-Transformer 在处理城市路网空间结构表示学习任务时具有显著的优势,特别是在捕捉复杂拓扑结构和语义信息方面表现出色。在成都数据集上,本文模型的准确率高达 88.10%,分别比 GCN、GAT 和 GraphSAGE 高出 0.62、1.08 和 0.98 个百分点;在武汉数据集上,本文模型与表现最佳的 GAT 准确率持平(均为 87.24%),同时,略优于 GCN 和 GraphSAGE;在郑州数据集上,尽管各编码器表现接近,但本文模型仍以 97.29% 的准确率领先;在贵阳数据集上,本文模型的优势相对明显,其准确率为 77.37%,分别比 GCN、GAT 和 GraphSAGE 高出 0.13、0.13 和 0.51 个百分点;天津数据集上,本文模型准确率为 90.25%,高于其他 3 种编码器;在太原数据集上,本文模型的准确率为 80.75%,优于 GCN、GAT 和 GraphSAGE;在青岛、重庆和长沙数据集上,本文模型的准确率分别为 92.83%、81.28% 和 85.21%,也都优于 GCN、GAT 和 GraphSAGE。

2.5.3 不同子图大小的结果对比

图 5 展示了子图大小对成都、武汉、太原、郑州、贵阳、天津 6 个城市数据集实验结果的影响。通过逐步调整子图大小从 2 至 20,观察到模型性能普遍呈现提升趋势,表现为准确率的持续上升。具体而言,各城市数据集的准确率提升幅度差异较大。其中,太原的准确率从 79.21% 提升至 80.75%,贵阳则从 76.10% 提升至 77.37%,两地的准确率提升幅度分别超过了 1.54 和 1.27 个百分点,是提升最为显著的城市。相比之下,武汉和成都的准确率虽然也有小幅提高,分别增长了 0.6 和 0.75 个百分点,但提升幅度相对

平稳。值得注意的是,郑州数据集中无论子图大小如何变化始终保持着较高的准确率,与其他城市相比具有明显优势。其主要原因是郑州的地理特征使其在空间结构上更具规律性,这样的结构可能使得模型更易捕捉到有效的空间关系。

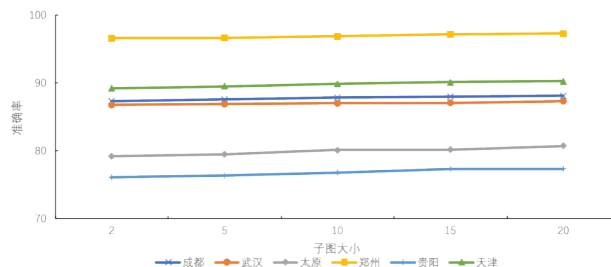


Fig. 5 Comparison of accuracy for different subgraph sizes

图 5 不同子图大小的准确率对比

本研究对不同子图大小下模型的训练时间进行了分析。实验结果如表 9 所示,其中,列出了在成都、武汉、郑州、贵阳、天津和太原 6 个数据集中,随着子图大小从 2 增加至 20,模型完成一个 epoch 所需的时间(单位:s)。实验分析表明,尽管子图大小的增加对运行时间有一定影响,但整体变化幅度较小,模型在处理不同大小子图时具有较高的稳定性。具体而言,成都和武汉的训练时间相对较长,其中,成都从 2 子图大小的 2.05 s 增加至 20 子图大小的 2.09 s,武汉则从 1.20 s 增加至 1.23 s。相比之下,郑州、贵阳、天津和太原的训练时间较短,且随子图大小增加的变化幅度较小。例如,郑州的训练时间从 0.54 s 增至 0.58 s,贵阳从 0.27 s 增至 0.29 s。实验结果表明,尽管模型在不同城市环境下的训练时间存在差异,但这种差异主要与城市的计算资源和环境有关,而非子图大小本身。

Table 9 Comparison of training time for different subgraph sizes

表 9 不同子图大小的训练时间对比

子图大小	成都	武汉	郑州	贵阳	天津	太原
2	2.05	1.20	0.54	0.27	0.92	0.53
5	2.06	1.20	0.54	0.27	0.92	0.53
10	2.06	1.21	0.56	0.28	0.93	0.54
15	2.08	1.23	0.57	0.28	0.93	0.55
20	2.09	1.23	0.58	0.29	0.93	0.55

3 结语

本文提出了一种基于 Graph-Transformer 的道路网络自监督学习模型,以便捕捉和表示道路网络的复杂拓扑结构隐含信息。模型框架由子图采样数据增强、Graph-Transformer 编码器和节点-子图对比损失 3 个部分组成。实验结果表明,本文模型在多个城市数据集上的准确率提升了 0.02%~2.45%,在多个城市数据集上的 F1 分数提升了 0.07%~2.55%,并且在下游任务社区划分效果上表现出更优的模块度 Q 值,提升了 0.02%~1.75%。通过消融实验,本

文模型架构、损失函数以及适当的子图大小组合能够进一步提升模型的性能。模型效果显著提升的深层次原因在于,模型中编码器节点注意力卷积NodeFormerConv的引入。通过自注意力机制以及核化变换,可以有效地捕捉节点间的复杂关系和全局结构信息,从而为城市路网的空间结构表示学习提供丰富的语义和拓扑特征。该编码器的优势在消融实验中也得到了验证,进一步证明了其在处理大规模复杂图结构数据时的优越性。

尽管模型在多个城市的数据集上表现优异,但未来的研究仍需包含立交桥和复杂十字路口等特征的路网结构中进行进一步验证。此外,模型性能的提升在很大程度上依赖于子图采样的有效性。目前采用的子图采样策略可能尚未完全覆盖所有类型的路网结构特征。因此,后续工作将重点优化子图采样策略,使其能够更有效地适应不同规模和复杂度的城市路网结构,同时,增强模型对全局结构特征的捕捉能力。下一步,将探索更高效的对比学习机制,尝试结合多视图对比学习或跨模态对比学习,进一步提升模型的代表学习能力。

#### 参考文献:

- [1] CHEN B F, WU Z F, HU W P. Complexity analysis of urban traffic road network based on fractal theory and GIS—taking Guangzhou urban area as an example[J]. *Tropical Geography*, 2011, 31(1): 46–51.  
陈斌峰,吴志峰,胡伟平. 基于分形理论与GIS的城区交通路网复杂性分析——以广州市城区为例[J]. *热带地理*, 2011, 31(1): 46–51.
- [2] ZHANG J, CHEN L. Reliability analysis of road network structure [J]. *Journal of Chang'an University (Natural Science Edition)*, 2010, 30(4): 77–81.  
张菁,陈荔. 道路网结构可靠性分析[J]. *长安大学学报(自然科学版)*, 2010, 30(4): 77–81.
- [3] PORTA S, CRUCITTI P, LATORA V. The network analysis of urban streets: a primal approach [J]. *Environment and Planning B: Planning and Design*, 2006, 33(5): 705–725.
- [4] JIANG B. A topological pattern of urban street networks: universality and peculiarity [J]. *Physica A: Statistical Mechanics and its Applications*, 2007, 384(2): 647–655.
- [5] DUAN Y, LU F. Robustness of city road networks at different granularities [J]. *Physica A: Statistical Mechanics and its Applications*, 2014, 411: 21–34.
- [6] JIANG B, DUAN Y, LU F, et al. Topological structure of urban street networks from the perspective of degree correlations [J]. *Environment and Planning B: Planning and Design*, 2014, 41(5): 813–828.
- [7] CASALI Y, HEINIMANN H R. A topological analysis of growth in the Zurich road network [J]. *Computers, Environment and Urban Systems*, 2019, 75: 244–253.
- [8] JIANG B, LIU C. Street-based topological representations and analyses for predicting traffic flow in GIS [J]. *International Journal of Geographical Information Science*, 2009, 23(9): 1119–1137.
- [9] MA D, OMER I, OSARAGI T, et al. Why topology matters in predicting human activities [J]. *Environment and Planning B: Urban Analytics and City Science*, 2019, 46(7): 1297–1313.
- [10] GAO S, WANG Y, GAO Y, et al. Understanding urban traffic-flow characteristics: a rethinking of betweenness centrality [J]. *Environment and Planning B: Planning and Design*, 2013, 40(1): 135–153.
- [11] JENELIUS E. Network structure and travel patterns: explaining the geographical disparities of road network vulnerability [J]. *Journal of Transport Geography*, 2009, 17(3): 234–244.
- [12] ORELLANA D, GUERRERO M L. Exploring the influence of road network structure on the spatial behaviour of cyclists using crowdsourced data [J]. *Environment and Planning B: Urban Analytics and City Science*, 2019, 46(7): 1314–1330.
- [13] ZHUANG D, WANG Q, ZHENG Y, et al. Advancing transportation mode share analysis with built environment: deep hybrid models with urban road network [DB/OL]. <https://arxiv.org/abs/2405.14079>.
- [14] ZHENG J, NI L M. Time-dependent trajectory regression on road networks via multi-task learning [C]//*Proceedings of the AAAI Conference on Artificial Intelligence*, 2013: 1048–1055.
- [15] FRUENSGAARD M, JEPSEN T S. Improving cost estimation models with estimation updates and road2vec: a feature learning framework for road networks [EB/OL]. <https://api.semanticscholar.org/CorpusID:44034092>.
- [16] LIU K, GAO S, QIU P, et al. Road2vec: measuring traffic interactions in urban road system from massive travel routes [J]. *ISPRS International Journal of Geo-Information*, 2017, 6(11): 321.
- [17] SHAHABI C, KOLAHDOUZAN M R, SHARIFZADEH M. A road network embedding technique for k-nearest neighbor search in moving object databases [C]//*Proceedings of the 10th ACM International Symposium on Advances in Geographic Information Systems*, 2002: 94–100.
- [18] LIU F, HO Y H, HUA K A. Privacy protected query processing with road network embedding [C]//*2011 IEEE International Conference on Advanced Information Networking and Applications*, 2011: 481–487.
- [19] JEPSEN T S, JENSEN C S, NIELSEN T D, et al. On network embedding for machine learning on road networks: a case study on the danish road network [C]//*2018 IEEE International Conference on Big Data*, 2018: 3422–3431.
- [20] 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.
- [21] 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.
- [22] TANG J, QU M, WANG M, et al. Line: large-scale information network embedding [C]//*Proceedings of the 24th International Conference on World Wide Web*, 2015: 1067–1077.
- [23] WANG D, CUI P, ZHU W. Structural deep network embedding [C]//*Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016: 1225–1234.
- [24] GOYAL P, FERRARA E. Graph embedding techniques, applications, and performance: a survey [J]. *Knowledge-Based Systems*, 2018, 151: 78–94.
- [25] XIE Y, GONG M, WANG S, et al. Sim2vec: node similarity preserving network embedding [J]. *Information Sciences*, 2019, 495: 37–51.
- [26] WANG M, LEE W C, FU T, et al. Learning embeddings of intersections on road networks [C]//*Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2019: 309–318.
- [27] HUANG S, SHAO C, LI J, et al. Feature extraction and representation of urban road networks based on travel routes [J]. *Sustainability*, 2020,

- 12(22): 9621.
- [28] JEPSEN T S, JENSEN C S, NIELSEN T D. Graph convolutional networks for road networks[C]//Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 2019: 460–463.
- [29] GHARAEI Z, KOWSHIK S, STROMANN O, et al. Graph representation learning for road type classification[J]. Pattern Recognition, 2021, 120: 108174.
- [30] NIPPANI A, LI D, JU H, et al. Graph neural networks for road safety modeling: datasets and evaluations for accident analysis [DB/OL]. <https://arxiv.org/abs/2311.00164>.
- [31] DWIVEDI V P, BRESSON X. A generalization of transformer networks to graphs[DB/OL]. <https://arxiv.org/abs/2012.09699>.
- [32] LI C, XIA L, REN X, et al. Graph transformer for recommendation[C]//Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2023: 1680–1689.
- [33] WU Q, ZHAO W, LI Z, et al. Nodeformer: a scalable graph structure learning transformer for node classification[J]. Advances in Neural Information Processing Systems, 2022, 35: 27387–27401.
- [34] JEPSEN T S, JENSEN C S, NIELSEN T D, et al. On network embedding for machine learning on road networks: a case study on the danish road network [C]//2018 IEEE International Conference on Big Data, 2018: 3422–3431.
- [35] LIU K, ZHANG M, XI G, et al. Enhancing fine-grained intra-urban dengue forecasting by integrating spatial interactions of human movements between urban regions [J]. PLoS Neglected Tropical Diseases, 2020, 14(12): e0008924.
- [36] SHEN Y, JIN C, HUA J, et al. TTPNet: a neural network for travel time prediction based on tensor decomposition and graph embedding [J]. IEEE Transactions on Knowledge and Data Engineering, 2020, 34(9): 4514–4526.
- [37] CHEN Y, LI X, CONG G, et al. Robust road network representation learning: when traffic patterns meet traveling semantics [C]//Proceedings of the 30th ACM International Conference on Information & Knowledge Management, 2021: 211–220.
- [38] BRIN S, PAGE L. The anatomy of a large-scale hypertextual web search engine [J]. Computer Networks and ISDN Systems, 1998, 30(1–7): 107–117.
- [39] JIAO Y, XIONG Y, ZHANG J, et al. Sub-graph contrast for scalable self-supervised graph representation learning [C]//2020 IEEE International Conference on Data Mining, 2020: 222–231.
- [40] NEWMAN M E J. Modularity and community structure in networks [J]. Proceedings of the National Academy of Sciences, 2006, 103(23): 8577–8582.
- [41] BOEING G. Global urban street network data. [EB/OL]. <https://doi.org/10.7910/DVN/3BZ3ZZ>.
- [42] KOU S H, YAO Y, ZHENG H, et al. Evaluation of urban traffic layout in China based on road network data and complex graph theory [J]. Journal of Geo-information Science, 2021, 23(5): 812–824.  
寇世浩,姚尧,郑泓,等. 基于路网数据和复杂图论的中国城市交通布局评价[J]. 地球信息科学学报, 2021, 23(5): 812–824.
- [43] THAKOOR S, TALLEC C, AZAR M G, et al. Bootstrapped representation learning on graphs [C]//ICLR 2021 Workshop on Geometrical and Topological Representation Learning, 2021.
- [44] KIPF T N, WELING M. Semi-supervised classification with graph convolutional networks [DB/OL]. <https://arxiv.org/abs/1609.02907>.
- [45] HAMILTON W, YING Z, LESKOVEC J. Inductive representation learning on large graphs [C]// Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017: 1025–1035.

(责任编辑:陈维捷)