

基于图组合优化的高效社区搜索

张安冉¹, 王兴芬^{1*}, 赵雨涵², 李立博¹

(1. 北京信息科技大学, 北京 102206; 2. 香港浸会大学, 香港 999077)

摘要: 针对大多数基于图神经网络(Graph Neural Network, GNN)的社区搜索方法中存在的时间开销巨大和“搭便车”效应问题, 本文提出一种基于图组合优化的高效社区搜索模型(Efficient Community Search Based on Graph Combinatorial Optimization, CS-ROMF). 该模型设计基于GNN的社区定位器来快速定位查询节点的潜在社区, 减少时间开销. 在此基础上设计基于强化学习(Reinforcement Learning, RL)的社区优化器调整候选社区的结构, 减轻“搭便车”效应. 在5个具有真实社区的数据集上进行大量实验, 结果表明CS-ROMF在所有评估指标上均优于基线模型. 其中, 相比结果最好的基线模型, CS-ROMF在 F_1 值、Jaccard值以及NMI上分别最高提升14.99%、20.67%和21.37%, 表明CS-ROMF减轻了“搭便车”效应. 同时, CS-ROMF能够显著提升搜索效率, 其运行速度比基于GNN的基线模型最多快10倍.

关键词: 社区搜索; 图神经网络; 图组合优化; 匹配策略; 强化学习; 社区检测

中图分类号: TP183; TP301

文献标识码: A

文章编号: 0372-2112(2025)02-0440-11

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.12263/DZXB.20240767

CS-ROMF: Efficient Community Search Based on Graph Combinatorial Optimization

ZHANG An-ran¹, WANG Xing-fen^{1*}, ZHAO Yu-han², LI Li-bo¹

(1. Beijing Information Science and Technology University, Beijing 102206, China;

2. Hong Kong Baptist University, Hongkong 999077, China)

Abstract: To address the significant time overhead and free-rider effect in most GNN-based community search methods, this paper proposes an efficient community search based on graph combinatorial optimization (CS-ROMF). CS-ROMF designs a GNN-based community locator to quickly pinpoint potential communities of the query nodes, thereby reducing time overhead. On this basis, CS-ROMF further designs an RL-based community optimizer to adjust the structure of candidate communities, mitigating the free-rider effect. Experiments conducted on five real-world datasets with true communities demonstrate that CS-ROMF outperforms advanced community search methods across all evaluation metrics. Specifically, compared to the best baseline model, CS-ROMF achieves maximum improvements of 14.99%, 20.67%, and 21.37% in F_1 score, Jaccard score, and NMI, respectively. Additionally, CS-ROMF can significantly improve search efficiency, running up to 10 times faster than the baseline model based on GNN.

Key words: community search; graph neural network; graph combinatorial optimization; matching strategy; reinforcement learning; community detection

1 引言

网络是数据挖掘领域中一个至关重要的研究对象, 它能够表示社会^[1]、自然^[2]和学术^[3]等领域的研究对象之间丰富的关系信息. 研究网络的一个关键步骤是识别和分析紧密相连的子图——社区. 现有的社区研究工作主要分为两类: (1) 社区检测^[4], 其目的是查找

网络中所有的社区, 该工作已经受到几十年的广泛研究; (2) 社区搜索^[5], 其目的是查找网络中包含查询节点的社区, 与社区检测不同, 社区搜索更加注重个性化搜索, 因此在近几年得到越来越多的关注.

在社区搜索中, 由于社区实体容易受到查询节点和网络结构的影响, 搜索合适的社区成为一项艰巨而富有挑战性的任务. 随着图神经网络^[6] (Graph Neural

Network, GNN)在各种图数据挖掘任务中取得巨大的成就,如图聚类^[7]、推荐系统^[8]、社区检测^[9]等,有研究提出利用GNN充分挖掘网络结构和节点特征信息,以交互的方式实现灵活的社区搜索^[10,11]. 这些研究取得了重大进展,然而大多数现有的方法在应用于真实世界中的网络时仍然面临以下挑战.(1)每进行一次查询就要重新训练整个模型,这种重新训练的过程造成巨大的时间开销,阻碍了社区搜索在现实场景中的应用.(2)忽略了社区研究中普遍存在的“搭便车”效应^[12](Free-rider Effect),导致最终得到的社区可能包含许多与查询节点无关的节点.

为了解决以上问题,本文提出一种基于图组合优化的高效社区搜索模型(Efficient Community Search Based on Graph Combinatorial Optimization, CS-ROMF). 该模型包含两个组成部分:社区定位器和社区优化器. 首先,社区定位器将输入的训练查询节点和相应的真实社区编码成向量,设计基于查询节点的最大相关社区和真实社区之间的相似性度量作为损失函数来进行训练. 对于新的查询节点,训练好的定位器利用最近邻匹配策略找到候选社区,实现快速定位. 值得注意的是,当给出多组查询时,社区定位器只需要训练一次模型. 其次,社区优化器通过引入全局结构模式来进一步调整候选社区中的节点成员,本文将这一优化的过程设计为基于深度强化学习(Reinforcement Learning, RL)的图组合优化(Graph Combinatorial Optimization, GCO)任务,并采用策略梯度训练这个过程. 对于已定位的候选社区,优化器提供两个动作:(1)添加与查询节点相关的外部节点;(2)删除与查询节点无关的内部节点,从而减轻“搭便车”效应.

本文的主要贡献如下:

(1)提出并分析了基于GNN的社区搜索存在的两个重要问题:时间开销巨大和“搭便车”效应. 为解决以上问题,提出了一种基于图组合优化的高效社区搜索模型——CS-ROMF,该模型包含两个关键组成部分:社区定位器和社区优化器.

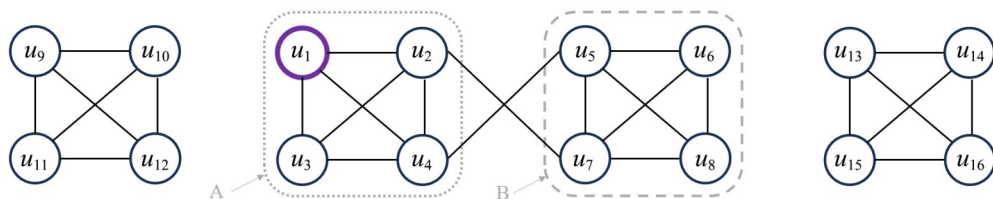


图1 “搭便车”效应示例图

定义3 最大相关社区(the Biggest Corresponding Community)给定一组查询节点 $V_q = \{v_1, v_2, \dots, v_n\}$, c^i 是节点 v_i 的 k 阶邻居子图,则查询节点 V_q 的最大相关社区为 $C^q = c^1 \cup c^2 \cup \dots \cup c^n$,如图2所示,其中 k 由查询节点

(2)社区定位器利用基于GNN的匹配策略初步定位与查询节点相关的潜在社区;社区优化器将社区搜索转换为图组合优化问题并利用RL来优化候选社区的结构.

(3)在具有真实社区的现实世界网络上对本文所提模型CS-ROMF和基线模型进行实验验证,实验结果证明了CS-ROMF的有效性和高效性.

2 背景知识和相关工作

2.1 背景知识

给定无向图 $G=(V, E, X)$,其中 V 是节点的集合, E 是边的集合, X 是节点特征矩阵. 如果一条边连接两个节点,则这两个节点是相邻的,并且 $N(v) = \{u \in V \mid \{u, v\} \in E\}$ 是节点 v 的邻居的集合,由此可得节点 v 的度为 $\text{deg}(v) = |N(v)|$. 无向图 G 中节点 u 和 v 之间的距离(最短路径长度)表示为 $\text{dist}(u, v)$.

问题定义(社区搜索, Community Search, CS)社区搜索的目标是根据查询请求,以在线的方式搜索高质量的社区^[13]. 具体来说,给出一个无向图 G 和一组查询节点的集合 $V_q \subseteq V$,其目的是找到包含 V_q 且满足以下性质的社区:(1)连通性,即社区中的节点是连通的;(2)内聚性,即社区中的节点是紧密相连的.

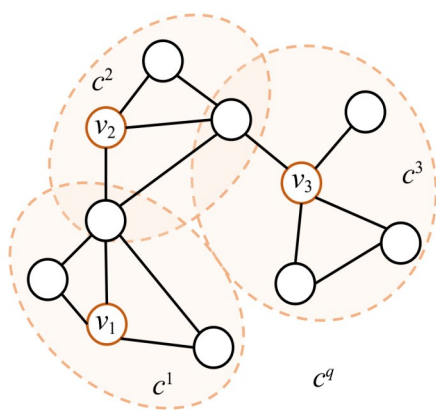
定义1 内聚性(Cohesiveness)通常利用内聚性度量来约束一个社区的紧密程度,例如 k -core,给定一个正整数 k ,如果社区 G_c 中的每一个节点 u 都满足 $\text{deg}(u) \geq k$,则社区 G_c 满足结构紧密性.

定义2 “搭便车”效应(Free-rider Effect)给定查询节点 V_q , C 为基于优化函数 f 识别到的社区, C^* 是最优解. 如果 $f(C \cup C^*) \geq f(C)$,则优化函数会受到“搭便车”效应的影响^[12]. 这种效应如图1所示,假设查询节点为 u_1 ,那么符合条件的社区应该是 A ,因为该社区内部连接密集,外部连接稀疏,社区内节点与查询节点紧密相关. 然而,一些相关研究将 $A \cup B$ 作为目标社区,当搜索包含 u_1 的社区时,子图 B 便成了一个“搭便车者”.

间的距离决定.

2.2 相关工作

社区搜索的目的是在大规模网络中找到包含给定查询节点的稠密子图,实现个性化的社区检测,该问题

图2 最大相关社区 C^q 示例

最早由 Sozio 等人^[5]提出. 随后, Fang 等人^[13]对现有的社区搜索方法进行了全面的调查研究, 基于不同的结构内聚性约束出现了 k -core^[14]、 k -truss^[15] 等各种社区搜索模型. 由于网络具有复杂的拓扑结构和节点特征, 传统的社区搜索方法无法灵活地利用这些复杂信息并充分地挖掘它们之间的关系^[13], 因此, 越来越多的学者致力于通过各种深度学习技术实现高效灵活的社区搜索.

2.2.1 基于 GNN 的社区搜索

GNN 通过将网络中的内容和结构特征编码成节点嵌入充分挖掘网络中的复杂信息, 用于分类等下游任务, GNN 已经在各种图数据挖掘任务中取得巨大成就. Liu 等人^[16]研究了 GNN 在社区检测中的作用, 总结了面临的挑战和机遇. 受此启发, Chen 等人^[11]首次提出基于 GNN 的社区搜索模型——ICS-GNN, 对社区进行迭代搜索, 在每次迭代中, 利用广度优先搜索 (BFS) 构建候选子图. Jiang 等人^[17]提出一个查询驱动模型——QD-GNN, 结合局部查询依赖和全局节点嵌入实现无参数的社区搜索. Li 等人^[18]利用 GNN 将结构和属性融合为统一的优化指标, 然后通过训练随机游走采样的局部子图进行社区搜索. 基于 GNN 的社区搜索方法有效地利用结构拓扑和内容特征等信息, 实现灵活搜索并提高搜索社区的性能, 还解决了参数敏感等问题^[10, 19]. 但是, 大多数基于 GNN 的模型每进行一次查询就会重新训练模型, 造成巨大的时间开销, 并且仍然面临“搭便车”效应问题的困扰.

2.2.2 基于 GNN 的子图匹配

子图匹配是指确定目标图中是否存在查询图的任务. 随着图神经网络的兴起, 基于 GNN 的图匹配方法^[20, 21]被提出, 并取得了巨大的成果. Wu 等人^[22]利用匹配策略实现社区顺序嵌入, 通过捕获社区之间的相似性找到网络中所有的潜在社区, 在很大程度上减少了时间开销. Zhang^[23]在时间图上寻找满足时态和拓扑

信息的匹配子图集. Nguyen 等人^[24]提出可解释的神经子图匹配, 引入图可学习多跳注意力网络 (Graph Learnable Multi-hop Attention Networks, GLeMA) 联合优化子图匹配和查找子图. 受以上研究的启发, 本文将匹配策略引入社区搜索任务, 实现潜在社区的快速定位.

2.2.3 基于 RL 的图组合优化

随着深度强化学习在各种优化任务中取得成功, 研究人员利用 RL 技术来解决 GCO 问题^[25-27]. S2V-DQN^[26]首次尝试使用深度 Q 学习 (DQN) 将 GCO 问题建模为顺序决策问题. 最近, Rizvee 等人^[28]将经过二次无约束二元优化 (Quadratic Unconstrained Binary Optimization, QUBO) 的汉密尔顿函数作为 RL 范式中的通用奖励函数, 以二元变量的形式对 GCO 问题进行建模. 而 Wang 等人^[29]则提出一种基于 BERT 的深度 RL 框架 BDRL, 针对特定任务进行微调, 从而更好地应用于 GCO 问题. 调整查询节点候选社区的节点成员也是一个 GCO 问题, 因此本文提出基于 RL 的社区优化器实现社区结构优化.

本文重点研究基于 GNN 的社区搜索存在的时间开销巨大和“搭便车”效应的问题, 引入匹配策略和图组合优化策略来设计解决方案.

3 CS-ROMF 框架

本节首先介绍 CS-ROMF 模型的整体框架描述, 然后进一步展开 CS-ROMF 的两个重要组成部分——社区定位器和社区优化器.

3.1 框架描述

为了更好地解决时间开销巨大和“搭便车”效应的问题, 本文提出一种基于图组合优化的高效社区搜索模型 CS-ROMF, 其基本框架如图 3 所示. CS-ROMF 的基本框架由社区定位器和社区优化器两个部分组成. (1) 社区定位器. 定位器对图 G 、训练的查询节点 Q_{train} 以及对应社区 C_{train} 进行编码, 通过计算查询节点的最大相关社区 C^{train} 与真实社区 C_{train} 的损失函数来训练定位器, 当给出新的查询节点 $Q = \{q_1, q_2, \dots, q_n\}$ 后, 定位器首先确定最大相关社区 C^q , 通过迭代训练, 得到候选社区 C_a . 为了提高效率, 可以直接将查询节点最大相关社区作为候选社区的初始社区. (2) 社区优化器. 考虑到定位器检测到的社区可能存在“搭便车者”, 引入社区优化器对候选社区的结构进行调整. 社区优化器采用深度 RL 的“状态”——“动作”——“奖励”机制, 将初始社区视为初始状态 S_0 , 通过自动学习启发式判别当前状态下的节点是否需要执行“删除”或“添加”动作. 每进行一次“动作”状态更新, 这一过程就会实现迭代地排除社区内的现有节点或向外边界 (社区内节点的邻居) 扩展. 这个迭代过程重复进行, 直到遇到“STOP”信号, 此时得到最终的预测社区 \hat{C} .

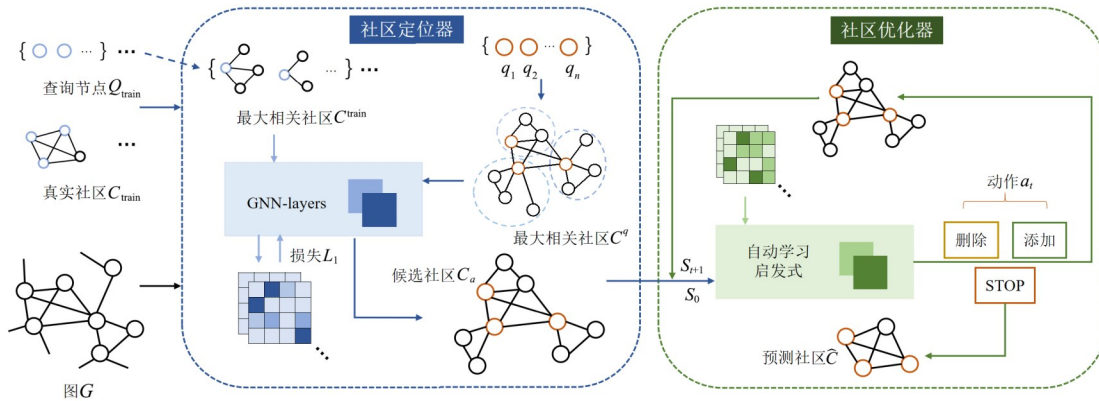


图3 CS-ROMF框架图

3.2 社区定位器

本节将介绍社区定位器的部分. 具体来说,在社区定位器中输入一组训练查询节点以及对应的社区,通过训练GNN模型来对其进行编码并最大化每组查询节点的最大相关社区中与其真实社区的相似性,从而拟合训练数据. 当给出一组新的查询节点时,社区定位器根据子图匹配来选择候选社区,通过这种方式,CS-ROMF可以快速定位查询节点的潜在社区. 算法1给出了社区定位的主要过程.

3.2.1 模型训练阶段

基于GNN的社区搜索模型通常直接将社区搜索建模为0/1分类问题,通过训练GNN模型获得图中所有节点的嵌入向量 $z(u)$ 以及节点属于某一社区的概率 $p(u)$. 在训练过程中,GNN作为编码器的基本组件,编码器通过 k 层GNN传播和聚合邻居信息,最终将前几层中的节点嵌入串接起来,得到节点 u 的最终嵌入 $z(u)$,如下所示:

$$z(u) = W^2 \cdot \text{CONCAT}(z^{(0)}(u), z^{(1)}(u), \dots, z^{(k)}(u)) \quad (1)$$

$$p(u) = \text{softmax}(z(u)) \quad (2)$$

其中, $z^{(l)}(u)$ 表示节点 u 在第 l 层网络中的节点嵌入, $W^l(l=1,2)$ 是权重参数. 然而,当存在多组查询节点时,每进行一次搜索,模型就要重新训练一次,这种重新训练的过程会产生巨大的时间开销. 因此,本文针对该问题提出了社区定位的思想,只需要训练一次模型,每组查询在训练好的模型上通过特定的定位来实现社区搜索.

给定训练的查询节点 $Q_{\text{train}} = \{Q_1, Q_2, \dots, Q_n\}$ 和相应的真实社区 $C_{\text{train}} = \{C_1, C_2, \dots, C_n\}$,可得查询节点的最大相关社区 $C^{\text{train}} = \{C^1, C^2, \dots, C^n\}$. 定位器对 $Q_{\text{train}}, C_{\text{train}}$ 和 C^{train} 分别进行编码. 对于网络中的节点 $u \in V$,对其进行特征增强, $x'(u) = [x(u), \text{deg}(u), \max(\text{DN}(u), \min(\text{DN}(u)), \text{mean}(\text{DN}(u)), \text{std}(\text{DN}(u)))]$,其中, $x(u)$ 是默认为1的节点 u 的原始特征, $\text{DN}(u) = \{\text{deg}(v) | v \in N(u)\}$. 这样可以提高节点特征的表达能,并且对于没有节点特征的网

算法1 社区定位

输入:图 $G=(V, E, X)$,训练样本 $Q_{\text{train}} = \{Q_1, Q_2, \dots, Q_n\}$,真实社区 $C_{\text{train}} = \{C_1, C_2, \dots, C_n\}$,查询节点组 Q_a ,阈值 γ .

输出:节点嵌入 $z(u)$,候选社区 C_a .

1. 计算 X 的增强特征矩阵 X'
2. 初始化编码 $z(u)$,设置 $V, C_a = \{\}$
3. WHILE 不收敛 DO
4. 生成 Q_i 的最大相关社区 C^i
5. FOR 每对 (C_i, C^i) DO
6. 编码 C_i, C^i 为 $z(C_i), z(C^i)$
7. 根据公式(7)计算损失函数
8. 更新 $z(u)$ 满足最小损失
9. 在 $z(C^i)$ 中选择每一个节点 v 的 $z(v)$
10. 根据公式(8)计算损失函数
11. 更新 $z(v)$ 满足最小损失
12. 生成 Q_a 最大相关社区 C_a'
13. 将 C_a' 中的节点 v 添加至节点集 V
14. WHILE V 非空 DO
15. 从节点集 V 选择一个节点 v
16. 编码 v 为 $z(v)$
17. FOR $z(v) \geq \gamma$ DO
18. $V \leftarrow V - \{v\}$
19. $C_a \leftarrow C_a \cup \{v\}$
20. RETURN $z(u), C_a$

络也是适用的. 原始节点经过线性变换更新为

$$z^{(0)}(u) = W^1 \cdot x'(u) \quad (3)$$

经过式(1)获得查询节点的嵌入向量,除此之外,还可以计算每个查询节点组 Q_i 对应真实社区 C_i 的嵌入表示:

$$z(C_i) = \sum_{u \in C_i} z(u) \quad (4)$$

当每个真实社区映射为一个 d 维向量时,要确保子图关系能够在嵌入空间中得到恰当的反映. 如果社区 C_a 是社区 C_b 的子图,那么对应的社区嵌入 $z(C_a)$ 和

$z(C_b)$ 必须满足:

$$z(C_a)[i] \leq z(C_b)[i], \forall i=1, \dots, d, \text{ if } C_a \subseteq C_b \quad (5)$$

对于每个查询节点组 Q_i 中的节点 q_j , 其 k -ego 网络的嵌入表示为 k 阶邻居节点的集合, 即 $z(C^{q_j}) = \sum_{v \in N_{q_j}^k} z(v)$,

由此可得 Q_i 的最大相关社区 C^j 的嵌入表示:

$$z(C^j) = \sum_{q_j \in Q_i} z(C^{q_j}) \quad (6)$$

当两个社区的嵌入非常接近时, 则认为这两个社区是相似的, 为了实现这个目标, 设计损失函数为

$$L_1 = \sum_{\text{Pos}} E(z(C_i) - z(C^j)) + \sum_{\text{Neg}} \max\{0, \alpha - E(z(C_i), z(C^j))\} \quad (7)$$

其中, $E(z(C_i), z(C^j)) = \|\max\{0, z(C_i) - z(C^j)\}\|_2^2$, Pos 表示正样本的集合, 即每对 (C_i, C^j) 中 C_i 是 C^j 的子图; Neg 表示负样本的集合, α 是距离参数.

为了更好地训练模型, 在每一次迭代中, 嵌入模型分别针对每组查询 Q_i 的真实社区 C_i 和最大相关社区 C^j 中的节点 v 输出 $z(C_i)$ 和 $z(v)$, 计算损失函数和模型参数的梯度. 在每次迭代结束后, 反向传播梯度并更新嵌入模型, 随着更新的参数, 嵌入模型移动到下一次迭代, 输出 $z(C_i)$ 和 $z(v)$, 计算损失并反向传播梯度直至收敛.

3.2.2 社区定位阶段

本文在社区定位阶段将定位过程构建为一个二分类预测问题, 对于每组查询节点, 不需要再重新训练模型, 直接执行社区定位过程.

假设 $y_{v_s} \in \{0, 1\}^n$ 表示每个查询节点组 Q_i 的最大相关社区 C^j 中的节点 v 的真值向量, $y_{v_s} = 1$ 当且仅当节点 v_s 满足 $v_s \in C_i$, 否则, $y_{v_s} = 0$, $\gamma \in [0, 1]$ 为预测参数, 则预测损失函数设计为 $z(v_s)$ 和 y_{v_s} 之间的二元交叉熵函数 (BCE), 优化损失函数表述如下:

$$L_2 = \sum_{Q_i \in Q_{\min}} \frac{1}{n} \sum_{s=1}^n - (y_{v_s} \log(z(v_s)) + (1 - y_{v_s}) \log(1 - z(v_s))) \quad (8)$$

在给出新的查询节点组 Q_a 时, 不需要重新训练模型就可以生成该查询节点组的候选社区 C_a . 首先找到查询节点组 Q_a 的最大相关网络 C'_a , 经过编码获得 C'_a 中所有节点的嵌入 $z(v_s)$, 根据广度优先搜索 (BFS) 遍历 C'_a 中的所有节点. 当访问节点 v_s 时, 如果 $z(v_s) \geq \gamma$, 则将节点 v_s 添加到候选社区 C_a 中, 遍历结束获得查询节点组 Q_a 的候选社区 C_a .

3.3 社区优化器

社区定位器将查询节点的最大相关社区作为候选社区, 这一假设在减少时间开销的同时牺牲了灵活性,

因此候选社区中可能存在“搭便车者”. 为了减轻“搭便车”效应, CS-ROMF 进一步设计社区优化器来调整社区结构并提供社区搜索的灵活性.

3.3.1 优化核心思想

给定一个查询节点组 Q_a 的候选社区 C_a , 为了优化社区结构, 优化器需要根据全局结构模式丢弃 C_a 中不相关的节点, 或者从 C_a 的外边界吸收相关节点, 从而输出更加精确的社区. 本文将这种社区结构优化问题看作图组合优化问题, 因此, 优化器需要在特定的节点集中选择合适的节点来优化目标. 社区优化器采用 RL 设计自动学习启发式来调整社区. 算法 2 给出了社区优化的主要过程.

算法 2 社区优化

输入: 图 $G=(V, E, X)$, 节点嵌入 $z(u)$, 候选社区 C_a

输出: 预测结果 \hat{C}

1. 初始化优化器为 A
2. WHILE 不收敛 DO
3. 生成训练样本集 D
4. FOR $(C^u \cup \partial C^u, \hat{C}^u)$ in D
5. 前馈传播 $C^u \cup \partial C^u$ 至 A 并获得 τ
6. 根据公式(15)更新 ϕ, ϕ
7. 传播 C_a 至 A 并获得 \hat{C}
8. RETURN \hat{C}

3.3.2 优化任务设计

社区结构优化的过程就是 RL 目标任务实现的过程. 将社区优化器的初始状态设置为 $S_0 = C_a \cup \partial C_a$, ∂C_a 是候选社区 C_a 的外边界. 在第 t 个状态中, 优化器获得当前预测社区 C_t 和它的外边界 ∂C_t , 在此基础上, 社区优化器进一步执行删除 C_t 中不相关节点的“删除”动作 a_t^{exclude} 或者添加 ∂C_a 中相关节点的“添加”动作 a_t^{expand} , 社区修改为 $C_{t+1} = C_t \setminus \{a_t^{\text{exclude}}\} \cup \{a_t^{\text{expand}}\}$, 社区优化器切换到下一个状态. 每个状态都是 C_t 和 ∂C_t 的组合表示, “删除”的作用空间是 C_t , “添加”的作用空间是 ∂C_t , 社区优化器通过奖励来决定执行“删除”还是“添加”.

状态 (State). 初始状态设置为 $S_0 = C_a \cup \partial C_a$, t 时刻的状态表示为 $S_t = C_t \cup \partial C_t$, $s_t(u)$ 是 S_t 状态下节点 u 的表示, S_0 状态下节点 u 的表示 $s_0(u)$ 由 $z(u)$ 得出:

$$s_0(u) = \text{CONCAT}(z(u), \mathbb{I}\{u \in C\}) \quad (9)$$

其中, $\mathbb{I}\{\text{条件}\}$ 是一个指示函数, 如果条件成立则返回 1, 否则为 0. 随着状态的切换, 每个节点可以进一步根据当前状态下邻域的信息来更新表示:

$$s'_t(u) = \text{GNN}(s_t(u), \{s_t(v) | v \in S_t\}) \quad (10)$$

$$s_{t+1}(u) = \text{CONCAT}(s'_t(u), \mathbb{I}\{u \in C_{t+1}\}) \quad (11)$$

动作(Action).状态之间的切换通过执行相应的动作来完成.为调整社区结构,社区优化器定义了“删除”和“添加”两个动作,“删除”表示从当前状态下的社区中丢弃不相关的节点,“添加”表示从当前状态下的外边界中吸收相关的节点.两种动作的实现如下:

$$a_t^{\text{exclude}} = \text{softmax}\left(\text{MLP}_\phi\left(\left\{s_t(u) \mid u \in C_t\right\}\right)\right) \quad (12)$$

$$a_t^{\text{expand}} = \text{softmax}\left(\text{MLP}_\phi\left(\left\{s_t(v) \mid v \in \partial C_t\right\}\right)\right) \quad (13)$$

其中, a_t^{exclude} 和 a_t^{expand} 表示衡量获取特定节点的概率,通过多层感知器(Multilayer Perceptrons, MLP)计算获得; ϕ 和 ϕ 分别表示 a_t^{exclude} 和 a_t^{expand} 动作下的可学习参数.另外,构造一个虚拟节点作为停止动作“STOP”学习的停止信号,该虚拟节点是一个具有全零特征的孤立节点.在执行动作过程中,如果 a_t^{exclude} 选择了虚拟节点,则停止执行“删除”,继续执行“添加”直到结束;反之, a_t^{expand} 选择了虚拟节点,则停止执行“添加”,继续执行“删除”直到结束.

奖励(Reward).调整社区结构的目的是通过执行动作来获得更加精确的社区,因此优化器选择任意一个评价指标(如 F_1 值)进行奖励计算.将经过动作 a_t 带来的 F_1 值之差作为奖励,定义为 r_t :

$$r_t = \delta(C_{t+1}, \hat{C}) - \delta(C_t, \hat{C}) \quad (14)$$

其中, δ 表示评估指标, \hat{C} 表示真实社区.

3.3.3 训练优化器

优化器采用策略梯度(Policy Gradient)方法训练可学习参数 ϕ 和 ϕ ,以“删除”为例,当给定候选社区 C_a ,“删除”动作轨迹为 $\tau = \{S_0, a_0, S_1, a_1, \dots, S_T, a_T, S_{T+1}\}$,随着动作沿着轨迹 τ 执行,可学习参数 ϕ 更新如下:

$$\phi \leftarrow \phi + l \sum_{i=1}^T \nabla_\phi \log(a_i | S_i) \cdot r_i \quad (15)$$

其中, l 表示参数 ϕ 的学习率.经过优化器的调整,得到最终的预测社区 \hat{C} .

3.4 算法分析

3.4.1 时间复杂度分析

社区定位器以GNN为底层框架,GNN对每个节点的邻居特征进行聚合,其时间开销为 $\sum_{i=u}^n d_u$, d_u 是节点 u 的度, n 是节点数,因此一般GNN的时间复杂度为 $O(|E|)$.社区定位器在生成最大相关社区时的时间复杂度均为 $O(1)$,计算损失迭代更新编码的复杂度在最坏情况下是 $O(|C|^2 + |V|^2)$.社区优化器的时间复杂度取决于RL的训练策略,其中,遍历样本集 D 的复杂度为 $O(|D|)$,单次前馈传播和参数更新的复杂度为 $O(f(|C^u| + |\partial C^u|) + t)$, t 是更新参数所需要的时间,传播 C_a 的复杂度为

$O(g(|C_a|))$.因此,CS-ROMF整体的时间复杂度可以记为 $O(|C|^2 + |V|^2 + f(|C^u| + |\partial C^u|) + g(|C_a|) + |E| + t)$.

3.4.2 空间复杂度分析

空间复杂度主要取决于输入图的规模、训练样本、真实社区和候选社区数量,因此,总体空间复杂度可以表示为 $O(|V| + |E| + |C_{\text{train}} + C_a| + |D|)$.

4 实验结果

4.1 实验环境

本实验所用计算机的配置如下:处理器为Intel(R) Xeon (R) Silver 4210 CPU @ 2.20 GHz,内存(RAM)256 GB,操作系统为Linux.另外,所有实验均使用Python编程实现.

4.2 数据集

为了全面评估模型的有效性,本文在单一数据集(具有部分标记社区的图)和混合数据集(多个不同单一数据集的组合)上进行实验.单一数据集从SNAP中选取3个包含重叠社区的常见现实网络:Amazon、DBLP和Livejournal.混合数据集通过将两个不同的网络组合获得,例如添加跨网络的连边,社区之间的内部连通性并不会受到干扰.所有数据集的统计特性如表1所示.

表1 数据集的统计特性

	#V	#E	#C	C_{Max}	C_{Avg}
Amazon	6 926	17 893	1 000	30	9.38
DBLP	37 020	149 501	1 000	16	8.37
Livejournal	69 860	911 179	1 000	30	13.00
Amazon*DBLP	43 946	172 394	2 000	30	8.88
DBLP*Livejournal	106 880	1 070 680	2 000	30	10.69

注:#V表示节点数量,#E表示边数量,#C表示社区数量, C_{Max} 表示最大社区规模, C_{Avg} 表示平均社区规模.

4.3 基线模型

为了评估模型的效率和有效性,本文将CS-ROMF与5种基线模型进行对比实验,其中 k -core和 k -truss是经典的基于结构约束的社区搜索算法,ICS-GNN、QD-GNN和CommunityAF是基于GNN的社区搜索模型,具体描述如下:

(1) k -core^[5]:基于 k -core的社区搜索算法是一种利用图的 k -core分解来进行社区搜索的方法. k -core是图 G 中包含查询节点 V_q 的一个最大子图,其中每个节点的度数至少为 k .

(2) k -truss^[13]:与 k -core不同,基于 k -truss的社区搜索算法要保证包含查询节点的最大子图满足其中的每条边至少存在于 $(k-2)$ 个三角形中.

(3)ICS-GNN^[11]:ICS-GNN是一种基于GNN的交互式社区搜索模型,利用用户标签信息指导GNN模型训

论,根据节点之间的相似度寻找社区.

(4) QD-GNN^[16]: QD-GNN 是一种查询驱动的 GNN 模型,结合了局部查询依赖结构和全局图嵌入.

(5) CommunityAF^[30]: CommunityAF 是一种基于实例的社区搜索方法,利用增量式 GNN 来处理具有社区感知结构特征的节点表示,并选择自回归流模型实现并行训练.

4.4 参数设置与评估指标

4.4.1 参数设置

对于 CS-ROMF,本文在社区定位器中使用两层 GCN 并设置嵌入大小固定为 64,使用 Adam 对参数进行优化,社区定位器的默认学习率为 0.000 1,默认批量大小为 32,社区优化器的默认学习率为 0.001. 对于基线模型,本文按照原文献给出的参数范围仔细设置,并在每一个数据集上进行调优. 根据文献[12],对于 k -core,本文默认取 $k=3$;对于 k -truss,本文默认取 $k=4$,因为 $(k+2)$ -truss 包含 k -core.

4.4.2 评估指标

对于具有真实社区的网路,最常用的评估指标是 F_1 值和 Jaccard 值. 此外,本文还选取了归一化互信息 (NMI) 作为补充度量. 所有评估指标计算的对象为预测社区 \hat{C} 的节点集合 \hat{V} 与真实社区 C 的节点集合 V ,则 F_1 值可以表示为

$$F_1 = \frac{2 \cdot p \cdot r}{p + r} \quad (16)$$

其中, $p = \frac{|\hat{V} \cap V|}{|\hat{V}|}$ 是精确率, $r = \frac{|\hat{V} \cap V|}{|V|}$ 是召回率.

Jaccard 值可以表示为

$$J = \frac{|\hat{V} \cap V|}{|\hat{V}| + |V| - |\hat{V} \cap V|} \quad (17)$$

NMI 可以表示为

$$\text{NMI} = \frac{I(\hat{C}, C)}{\sqrt{H(\hat{C}) \cdot H(C)}} \quad (18)$$

其中, $I(\hat{C}, C)$ 是预测社区与真实社区之间的互信息, H 和 \hat{H} 分别是真实社区和预测社区的熵,定义如下:

$$I(\hat{C}, C) = \sum_{i=1}^{|\hat{V}|} \sum_{j=1}^{|\hat{V}|} P(i, j) \log \frac{P(i, j)}{P(i)P(j)} \quad (19)$$

$$H(C) = - \sum_{i=1}^{|\hat{V}|} P(i) \log P(i) \quad (20)$$

$$H(\hat{C}) = - \sum_{j=1}^{|\hat{V}|} P(j) \log P(j) \quad (21)$$

其中, $P(i, j)$ 是真实社区和预测社区中节点的联合概率, $P(i)$ 和 $P(j)$ 分别是真实社区和预测社区中节点的边际概率.

4.5 性能比较以及分析

4.5.1 总体性能

本节通过对比 CS-ROMF 与 5 个基线模型在 5 个具有真实社区的数据集上的 3 种评估指标来验证 CS-ROMF 模型的有效性. 表 2 展示了 CS-ROMF 与基线模型比较的性能总结,通过观察可以得出:

(1) CS-ROMF 模型的有效性. 对于所有数据集上的所有评估指标,CS-ROMF 优于所有的基线模型,获得最佳结果. 特别地,与最佳基线模型相比,CS-ROMF 在 F_1 值、Jaccard 值和 NMI 上分别最高提升了 14.99%、20.67% 和 21.37%,证明了其出色的性能.

表 2 CS-ROMF 与基线模型比较的性能总结 (最佳结果通过加粗表示,次优结果通过下划线表示)

	Dataset	k -core	k -truss	ICS-GNN	QD-GNN	CommunityAF	CS-ROMF
F_1	Amazon	0.325 1	0.362 8	0.510 4	0.623 9	<u>0.772 1</u>	0.894 9
	DBLP	0.200 7	0.254 9	0.451 7	0.536 0	<u>0.593 7</u>	0.639 2
	Livejournal	0.197 4	0.212 8	0.429 6	0.503 1	<u>0.574 0</u>	0.637 9
	Amazon*DBLP	0.289 1	0.317 4	0.480 5	0.593 2	<u>0.611 9</u>	0.750 2
	DBLP*Livejournal	0.114 5	0.109 3	0.335 4	<u>0.390 1</u>	0.373 6	0.540 0
Jaccard	Amazon	0.237 9	0.240 5	0.458 2	0.524 9	<u>0.620 8</u>	0.827 5
	DBLP	0.115 1	0.130 6	0.361 4	<u>0.455 7</u>	0.430 3	0.514 7
	Livejournal	0.082 2	0.127 4	0.310 9	0.409 5	<u>0.417 9</u>	0.531 6
	Amazon*DBLP	0.142 9	0.201 5	0.366 1	0.472 8	<u>0.538 6</u>	0.615 4
	DBLP*Livejournal	0.027 4	0.013 9	0.241 6	0.285 7	<u>0.309 2</u>	0.470 1
NMI	Amazon	0.190 0	0.203 1	0.417 2	0.475 3	<u>0.569 9</u>	0.783 6
	DBLP	0.073 3	0.069 0	0.292 5	<u>0.384 4</u>	0.375 1	0.437 9
	Livejournal	0.020 6	0.025 8	0.247 1	0.333 5	<u>0.365 0</u>	0.432 7
	Amazon*DBLP	0.122 0	0.168 5	0.353 7	0.431 9	<u>0.487 5</u>	0.566 1
	DBLP*Livejournal	0.000 4	0.001 9	0.205 7	0.236 5	<u>0.280 6</u>	0.445 1

(2) CS-ROMF 模型可以减轻“搭便车”效应. 由于混合数据集中节点类型多样, 基线模型在搜索包含查询节点的特定类型的社区时具有不适应性, 搜索时易出现“搭便车”效应, 从而导致性能降低. 相较之下, 本文提出的 CS-ROMF 在混合数据集上的改进显著. CS-ROMF 先通过社区定位器精准定位到目标候选社区, 又通过社区优化器调整社区结构, 实验结果表明其在精确搜索高质量社区方面具有优越性.

(3) CS-ROMF 模型具有可扩展性和鲁棒性. 随着数据集规模的增大, 社区结构复杂性提高, 基线模型的性能严重降低. 传统的基于结构约束的模型 k -core 和 k -truss 的性能最低降到 0.04% 和 0.19%, 这暴露了基于结构约束的模型在处理大规模复杂数据集时可扩展性和鲁棒性较差; 虽然 ICS-GNN、QD-GNN 和 CommunityAF 在一定程度上解决了规模扩大和复杂性提高带来的挑战, 但是其性能也出现大幅度降低. 本文提出的 CS-ROMF 首先从局部角度针对查询节点进行定位, 受数据

集规模扩大的影响较小, 又利用全局结构模式进行优化, 减轻了数据集中复杂结构带来的扰动, 展现出优秀的性能, 表明了其具有良好的可扩展性和鲁棒性.

4.5.2 CS-ROMF 模型的时间开销

本文通过直接比较给出多组查询节点时 CS-ROMF 与所有基线模型的总运行时间来评估 CS-ROMF 模型的时间开销. 图 4 展示了所有模型在各个数据集上的性能 (F_1 值) 和运行时间 (s). 由图可知, 几乎在所有数据集上, CS-ROMF 比 ICS-GNN、QD-GNN 和 CommunityAF 耗时都更少. 通常, 大多数基于 GNN 的社区搜索模型在每次查询之前都要重新训练模型, 造成了巨大的时间开销. 而 CS-ROMF 对于多次查询仅需训练一次, 并通过社区定位器快速定位潜在社区, 从而减小时间开销, 提高效率. 值得注意的是, 即使在规模最大的数据集上, CS-ROMF 的运行时间也不超过 10 s; 同时, CS-ROMF 的性能 (F_1 值) 优于其他速度更快的模型, 这表明 CS-ROMF 的高效性.

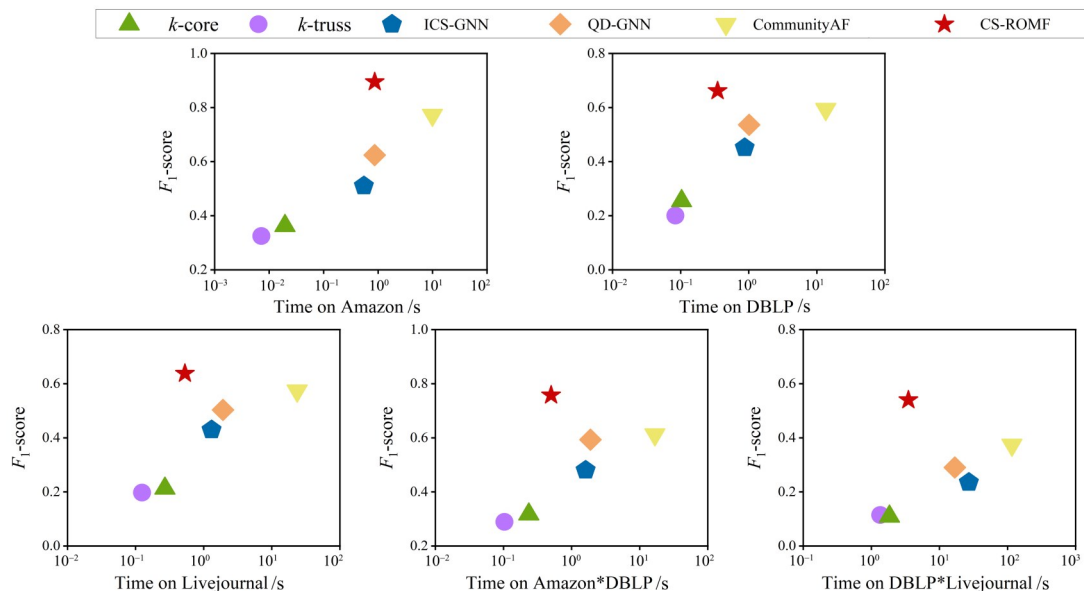


图 4 CS-ROMF 与基线模型的效率比较

4.5.3 参数敏感性

本节对 CS-ROMF 的参数敏感性进行讨论. 在数据集 DBLP*Livejournal 上进行实验, 通过改变训练轮次、嵌入维度、学习率和丢失率来观察评估指标的变化情况, 实验结果如图 5 所示. 由图 5(a) 和图 5(d) 可知, 随着训练轮次和丢失率的增加, 评估指标呈现出缓慢上升, 随后收敛; 由图 5(b) 可知, 当嵌入维度由 16 增加至 32 时, 评估指标略微下降, 随着嵌入维度的继续增加, 评估指标上升至原来水平并收敛; 由图 5(c) 可知, 学习率的改变对模型的影响微乎其微. 通过实验结果, 验证了 CS-ROMF 对于参数的改变并不敏感, 因此, CS-

ROMF 具有很好的鲁棒性.

4.5.4 减轻“搭便车”效应的案例研究

本文在混合数据集 Amazon*DBLP 上进行了案例研究, 以验证 CS-ROMF 可以有效减轻“搭便车”效应. 为了直观地展现“搭便车”效应, 分别选择位于 Amazon 与 DBLP 连接边缘的属于 Amazon 的节点 $Q_A = \{a_1, a_2, a_3\}$ 和属于 DBLP 的节点 $Q_D = \{d_1, d_2\}$ 作为两组查询节点, 搜索得到的两组查询节点的目标社区为 C_1 和 C_2 , 图 6 是 ICS-GNN 与 CS-ROMF 实验结果的可视化. 由图 6 可知, 无论给出哪组查询节点, ICS-GNN 的目标社区中均包含另一个数据集中的节点, 说明 ICS-GNN 并没有考虑两种数据

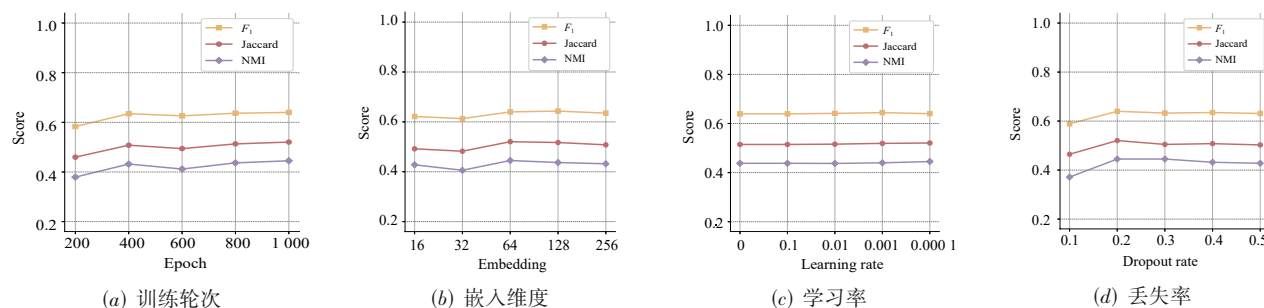


图5 参数对CS-ROMF的影响

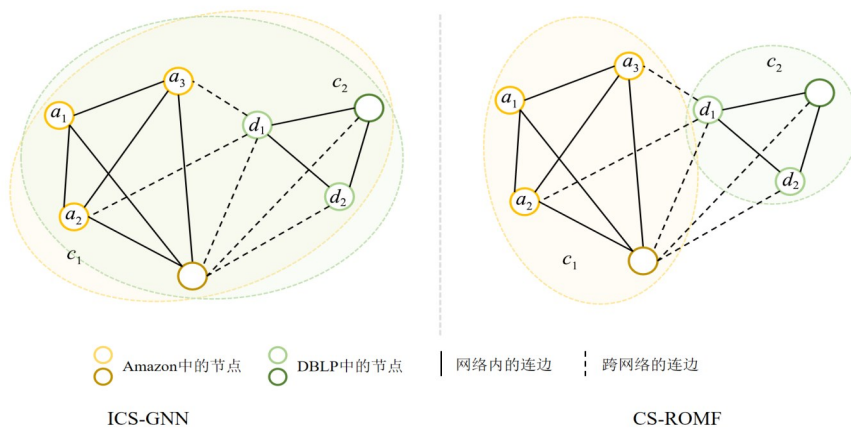


图6 CS-ROMF减轻“搭便车”效应的案例研究

集分别对应不同类型的节点,反而在训练时一致对待,导致目标社区中存在与查询节点并不相关的“搭便车者”。而CS-ROMF的目标社区内的节点均属于同一个数据集,所得的社区包含所有查询节点并具有结构紧密性,表明CS-ROMF有效减轻了“搭便车”效应。

4.5.5 CS-ROMF 模型的消融实验

为了评估CS-ROMF的社区定位器和社区优化器的有效性,本节进行了消融实验,实验结果如表3所示,由于 F_1 值和Jaccard值与NMI具有相似的趋势,表中省略

了它们的结果。

通过观察,相比随机选择 k 阶自我中心网络,社区定位器已经取得了显著的改进,并且对比表2中的结果,定位器也优于大多数基线模型,这表明社区定位器已经获得优异的性能,证明了定位目标社区的有效性。此外,对比CL和CS-ROMF发现,为模型引入社区优化器可以获得更好的性能。社区优化器对候选社区进行微调,CS-ROMF将这一微调的过程构建为基于RL的图组合优化任务,从而进一步提升了模型的性能。

表3 消融实验

	Amazon	DBLP	Livejournal	Amazon*DBLP	DBLP*Livejournal
k -ego subgraph	0.426 5	0.318 9	0.292 1	0.378 4	0.246 7
CL	0.701 0	0.437 9	0.403 5	0.566 1	0.421 9
CS-ROMF	0.786 3	0.462 9	0.432 7	0.580 9	0.445 1

注: k -ego subgraph是以 k 阶自我网络的形式随机选择一些子图生成的,CL表示由社区定位器检测到的候选社区,CS-ROMF是通过优化定位器检测到的候选社区获得的。

4.6 CS-ROMF 模型的适用场景

本文提出的CS-ROMF是一种聚焦于特定查询节点的图分析方法,即使在大规模网络中,也可以实现快速精准的社区搜索,这对于现实世界场景具有重要的应用价值。以下将讨论CS-ROMF模型的两个适用场景。

4.6.1 社交网络分析

信息化和社交媒体高速发展使社交网络变得易于访问,人们使用这些平台进行互动和共享信息。一般来说,具有某种共同点的个体更有可能进行互动,社交网络中用户之间的频繁互动形成了模块化结构,即社区。

社区结构揭示了社交网络中用户之间的潜在联系和互动模式^[31]。例如,在社交网络中,借助 CS-ROMF 模型针对特定人员进行个性化的相关朋友推荐或兴趣偏好推荐。此外,CS-ROMF 对于理解社交网络中的信息传播、用户行为分析、影响力最大化等也具有重要意义。

4.6.2 欺诈检测

随着人工智能技术的日益普及,金融、保险、医疗等各种平台越来越容易受到欺诈攻击,欺诈者往往利用各种通信手段实施团伙作案^[32]。要想在这些复杂网络中寻找与特定嫌疑节点相关的可疑群体,就迫切需要对其中的社区结构进行针对性搜索。当给出特定嫌疑节点时,可以借助 CS-ROMF 模型实现快速定位和精准划分,在一定程度上节省了时间和人力。此外,CS-ROMF 还可以用来定位潜在的威胁和恶意活动,从而达到风险管理的目的,也可以对网络中的异常用户群体进行实时监控和检测。

5 结论

本文的主要贡献是针对现有基于 GNN 的社区搜索方法训练耗时和“搭便车”效应两个问题,提出了一种基于图组合优化的高效社区搜索模型 CS-ROMF。本文的创新点在于设计的 CS-ROMF 模型包括两个重要组成部分:社区定位器和社区优化器。社区定位器通过训练 GNN 模型拟合训练数据,并利用匹配策略快速定位查询节点的潜在社区,减少时间开销。社区优化器在定位器的基础上进一步调整候选社区的结构得到最终目标社区,缓解“搭便车”效应。基于真实数据集的实验结果表明,在模型有效性方面,CS-ROMF 有效地减轻了“搭便车”效应,并且优于现有的社区搜索方法。在模型效率方面,CS-ROMF 模型的运行时间普遍低于基于 GNN 的基线模型。在未来的工作中,可以进一步研究更加新颖的图组合优化策略,以提高社区搜索的效率,探索图大模型与社区搜索之间的关系。

参考文献

- [1] 杜明, 宋嘉祎, 周军锋. 规模受限的影响力社区搜索[J]. 电子学报, 2023, 51(5): 1207-1214.
DU M, SONG J Y, ZHOU J F. Size-constrained influential community search[J]. Acta Electronica Sinica, 2023, 51(5): 1207-1214. (in Chinese)
- [2] HU A L, CHAN K C C. Utilizing both topological and attribute information for protein complex identification in PPI networks[J]. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2013, 10(3): 780-792.
- [3] LIU Y F, LIU Z, FENG X D, et al. Robust attributed network embedding preserving community information[C]// 2022 IEEE 38th International Conference on Data Engineering (ICDE). Piscataway: IEEE, 2022: 1874-1886.
- [4] SU X, XUE S, LIU F Z, et al. A comprehensive survey on community detection with deep learning[J]. IEEE Transactions on Neural Networks and Learning Systems, 2024, 35(4): 4682-4702.
- [5] SOZIO M, GIONIS A. The community-search problem and how to plan a successful cocktail party[C]//Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2010: 939-948.
- [6] TANG J B, YANG Y H, WEI W, et al. HiGPT: Heterogeneous graph language model[C]//Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. New York: ACM, 2024: 2842-2853.
- [7] FETTAL C, LABIOD L, NADIF M. Scalable attributed-graph subspace clustering[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2023, 37(6): 7559-7567.
- [8] ZHAO Y H, CHEN R, LAI R W, et al. Augmented negative sampling for collaborative filtering[C]//Proceedings of the 17th ACM Conference on Recommender Systems. New York: ACM, 2023: 256-266.
- [9] SHAN D B, DU X H, WANG W J, et al. KPI-HGNN: Key provenance identification based on a heterogeneous graph neural network for big data access control[J]. Information Sciences, 2024, 659: 120059.
- [10] CHEN J Z, GAO J. VICS-GNN: A visual interactive system for community search via graph neural network[C]// 2022 IEEE 38th International Conference on Data Engineering (ICDE). Piscataway: IEEE, 2022: 3150-3153.
- [11] CHEN J Z, GAO J, CUI B. ICS-GNN+: Lightweight interactive community search via graph neural network[J]. The VLDB Journal, 2023, 32(2): 447-467.
- [12] KIM J, LUO S Q, CONG G, et al. DMCS: Density modularity based community search[C]//Proceedings of the 2022 International Conference on Management of Data. New York: ACM, 2022: 889-903.
- [13] FANG Y X, HUANG X, QIN L, et al. A survey of community search over big graphs[J]. The VLDB Journal, 2020, 29(1): 353-392.
- [14] 张琦, 程苗苗, 李荣华, 等. 基于邻域 k-核的社区模型与查询算法[J]. 软件学报, 2024, 35(3): 1051-1073.
ZHANG Q, CHENG M M, LI R H, et al. Community model and query algorithm based on neighborhood k-core[J]. Journal of Software, 2024, 35(3): 1051-1073. (in Chinese)
- [15] LIAO X K, LIU Q, HUANG X, et al. Truss-based community search over streaming directed graphs[J]. Proceedings of the VLDB Endowment, 2024, 17(8): 1816-1829.
- [16] LIU F Z, XUE S, WU J, et al. Deep learning for community detection: Progress, challenges and opportunities[EB/OL]. (2020-05-17)[2024-08-19]. <https://arxiv.org/abs/2005.08225v2>.
- [17] JIANG Y L, RONG Y, CHENG H, et al. Query driven-

- graph neural networks for community search: From non-attributed, attributed, to interactive attributed[EB/OL]. (2021-04-08)[2024-08-19]. <https://arxiv.org/abs/2104.03583v2>.
- [18] LI Y, CHEN X X, ZHAO Y H, et al. Self-training GNN-based community search in large attributed heterogeneous information networks[C]//2024 IEEE 40th International Conference on Data Engineering (ICDE). Piscataway: IEEE, 2024: 2765-2778.
- [19] HASHEMI F, BEHROUZ A, REZAEI HAJIDEHI M. CS-TGN: Community search via temporal graph neural networks[C]//Companion Proceedings of the ACM Web Conference 2023. New York: ACM, 2023: 1196-1203.
- [20] LI Y J, GU C J, DULLIEN T, et al. Graph matching networks for learning the similarity of graph structured objects[EB/OL]. (2019-04-29)[2024-08-19]. <https://arxiv.org/abs/1904.12787v2>.
- [21] BAI Y S, DING H, BIAN S, et al. SimGNN: A neural network approach to fast graph similarity computation[C]//Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. New York: ACM, 2019: 384-392.
- [22] WU X X, XIONG Y, ZHANG Y, et al. CLARE: A semi-supervised community detection algorithm[C]//Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. New York: ACM, 2022: 2059-2069.
- [23] ZHANG H L. Temporal subgraph matching method for multi-connected temporal graph[J]. Information Sciences, 2025, 686: 121320.
- [24] NGUYEN D Q, TOAN NGUYEN T, JO J, et al. Explainable neural subgraph matching with learnable multi-hop attention[J]. IEEE Access, 2024, 12: 130474-130492.
- [25] SHAN C H, SHEN Y F, ZHANG Y, et al. Reinforcement learning enhanced explainer for graph neural networks[J]. Advances in Neural Information Processing Systems, 2021, 34: 22523-22533.
- [26] DUCHI J, TARLOW D, ELIDAN G, et al. Using combinatorial optimization within max-product belief propagation[M]//Advances in Neural Information Processing Systems 19. Cambridge, MA: The MIT Press, 2007: 369-376.
- [27] ZHANG Y, XIONG Y, YE Y, et al. SEAL: Learning heuristics for community detection with generative adversarial networks[C]//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM, 2020: 1103-1113.
- [28] RIZVEE R A, HASSAN R, KHAN M M. A graph neural network-based QUBO-formulated Hamiltonian-inspired loss function for combinatorial optimization using reinforcement learning[EB/OL]. (2023-11-27)[2024-08-19]. <https://arxiv.org/abs/2311.16277v1>.
- [29] WANG Q, LAI K H, TANG C L. Solving combinatorial optimization problems over graphs with BERT-Based Deep Reinforcement Learning[J]. Information Sciences, 2023, 619: 930-946.
- [30] CHEN J Z, XIA Y K, GAO J. CommunityAF: An example-based community search method via autoregressive flow[J]. Proceedings of the VLDB Endowment, 2023, 16(10): 2565-2577.
- [31] NIU Y D, LI Y C, KARRAS P, et al. Discovering personalized characteristic communities in attributed graphs[C]//2024 IEEE 40th International Conference on Data Engineering (ICDE). Piscataway: IEEE, 2024: 2834-2847.
- [32] YU J K, WANG H C, WANG X Y, et al. Group-based fraud detection network on e-commerce platforms[C]//Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. New York: ACM, 2023: 5463-5475.

作者简介



张安冉 女,1999年1月出生.现为北京信息科技大学网络空间安全专业硕士研究生.主要研究方向为数据挖掘与复杂网络.

E-mail: 2022020934@bistu.edu.cn



王兴芬 女,1968年9月出生.2005年毕业于哈尔滨工业大学管理学院.现任北京信息科技大学党委常委、副校长、教授、博士生导师.主要研究方向为大数据及商务分析、信息管理与智能决策等.

E-mail: xfwang@bistu.edu.cn



赵雨涵 男,1999年10月出生.现为香港浸会大学博士研究生.主要研究方向为数据挖掘与推荐系统.

E-mail: esyhzha@comp.hkbu.edu.hk



李立博 男,1999年3月出生.现为北京信息科技大学管理科学与工程专业硕士研究生.主要研究方向为大数据挖掘与分析、时间序列预测.

E-mail: 2022020958@bistu.edu.cn