

DOI: 10.13718/j.cnki.xdzk.2022.09.018

一种基于语义推理的网络社区发现模型

任薇¹, 阮淇昱¹, 韩孟凯¹, 邱玉辉²

1. 西南大学 计算机与信息科学学院 软件学院, 重庆 400715; 2. 西南大学 人工智能学院, 重庆 400715

摘要: 网络社区作为新的资源发布和共享方式, 其形成模式、发现方式等是网络发展研究热点之一。由于聚类的随机性以及现有划分算法对个体和链接属性信息的语义利用不充分, 社会网络的社区不能得到准确的分类。本文提出基于语义推理的网络社区发现模型, 通过节点关系的拓扑结构和节点间的语义联系, 抽取多层语义进行搜索并进行社区发现。在 ego-Facebook 数据集上的实验结果表明, 当社区大小增长到 1 000 以后, 本算法更加稳定, 更适用于节点包含丰富的语义且结构稀疏的网络。

关键词: 社区发现; 语义网络; 语义链; 多层模型; 知识图谱

中图分类号: TP393 **文献标志码:** A

文章编号: 1673-9868(2022)09-0178-09

开放科学(资源服务)标识码(OSID):



Semantic Link Based Cyber Community Discovery Model

REN Wei¹, RUAN Yuqi¹, HAN Mengkai¹, QIU Yuhui²

1. School of Computer and Information Science, Southwest University, Chongqing 400715, China;

2. School of Artificial Intelligence, Southwest University, Chongqing 400715, China

Abstract: As a new way of publishing and sharing resources, the network community is one of the hot spots in the research of network growth. Due to the randomness of clustering and the insufficient semantic utilization of individual node as well as link attributes by existing partitioning algorithms, communities in social networks cannot achieve satisfied classifications. This paper proposed a network community discovery model based on semantic reasoning, which extracts multi-layer semantics through the topology of network and the semantic connections between nodes. Experiment results on the ego-Facebook dataset show that the algorithm is more stable and more suitable for networks with rich semantics and sparse structure when the community size grows to more than 1 000.

Key words: community discovery; semantic network; semantic chain; multilayer model; knowledge map

社区发现(Community Discovery)是将复杂网络拓扑结构分解为有意义的节点集群的任务^[1]。目前主流社区发现方法基于聚类计算, 通过无监督学习训练发现共同群体^[2-3]。然而, 由于现有网络节点属性不同, 对于社区的定义各种各样, 聚类结果带有一定的随机性, 不能实现精准分类^[4]。另一方面, 现有划分算法只关注平面数据, 没有充分利用个体和链接属性信息的语义, 导致划分不准确^[5-6]。同时, 当前社区发现

算法主要通过将有向图转化为无向图来进行社区发现,此过程可能丢失很多细节导致社区结果划分不准确.为实现对节点关系的准确描述,在社区网络中加入有向性^[7],通过描述不同种类的链接,可以更真实模拟社交网络等现实世界网络.

Satuluri 算法^[8]与 LSW-OCD 算法^[9]都是根据节点的矢量将有向图转化为带方向权值的无向图,虽然 Satuluri 算法的复杂度已经在 LSW-OCD 得到较大改善,但节点中隐含着的语义与语义的关系却没有在算法中发挥作用.语义研究不仅关注事物概念的含义,还关注含义间的关系.在社会网络中,用户的行为都与用户本身的特征、爱好、习惯等紧密相连.社区发现中语义的引入,为挖掘非数据信息提供了可能,从而支持对网络社区更精确地划分.基于语义对社区发现的应用潜力,本文提出以语义推理为基础的网络社区发现模型.

1 语义网络及语义搜索

语义网络是当前万维网的扩展.在语义网络中,信息被赋予了明确的含义,使计算机和人能够更好合作^[10].语义网络中的概念节点按照层次进行组织,可以表现每个层次中不同节点之间的平面关系以及不同层次中节点的纵向关系^[11].

语义搜索是语义网络的核心^[12-14].语义搜索过程依据对本体处理原理的差异,可以分为3种:增强型语义搜索、知识型语义搜索及其他搜索.我们提出基于语义关系的发现模型,通过关注节点间的语义特性[相等(Equal)、相似(Similar)、引用(Reference)、序列(Sequence)、子类(Subclass)、蕴含(Implication)],采用语义链进行增强型推理,期望得到更贴近真实情况的社区发现结果.

2 基于语义关系推理的立体社区发现模型

人类的社交网络与其他网络不同,具有语义性,基于语义推理的社区发现更贴合真实情况.在社区结构划分的研究中,研究者主要关注两方面问题:一是社区的边界性,即社区边界必须是封闭的;二是社区的内部紧密性,即一个拓扑结构即使看上去有封闭的边界,但如果内部无紧密连接,也不能认定为社区.本文用总权值度量的方法衡量封闭性,用节点相似度来考察社区的紧密性.社区内部节点相似度的平均值通常大于社区间的相似度平均值.因此,本文给出社区结构定义如下:

给定一个混合图 G , 将 G 划分为 n 个子图 G_1, G_2, \dots, G_n , 使得任一子图 G_i 满足下列条件, 则 G_i 为网络 G 中的一个社区:

1) 对任意的节点 N , $d_{in}(N)$ 是节点 v 在 G_i 中的度(权值), $d_{out}(N)$ 是节点 v 与除 G_i 外的其他子图间的连接的总度, 则对于其他子图 G_j , 都有 $d_{out}(N) - d_{in}(N) \leq 0$;

2) N_i 的内部节点平均相似度大于其与其他子图间的节点平均相似度.

以上定义的社区结构涵盖了不同尺度的社区.

2.1 多层模型及社区发现模型

本文提出一种基于语义的网络社区发现、资源追踪的立体空间模型,在传统的平面二维网络模型基础上,针对语义网络节点间语义关系的特性,提出三维的空间立体模型(图1).该立体空间模型将传统意义上的网络结构按语义关系类型重新构建,模型拓扑结构分为纵面树拓扑结构(图2a)和平面图拓扑结构(图2b).

模型采用立体空间节点拓扑结构,利用节点语义及语义关系进行集成.平面图中和纵面树中的语义节点,其相互关系只能是特定的语义链中的一种

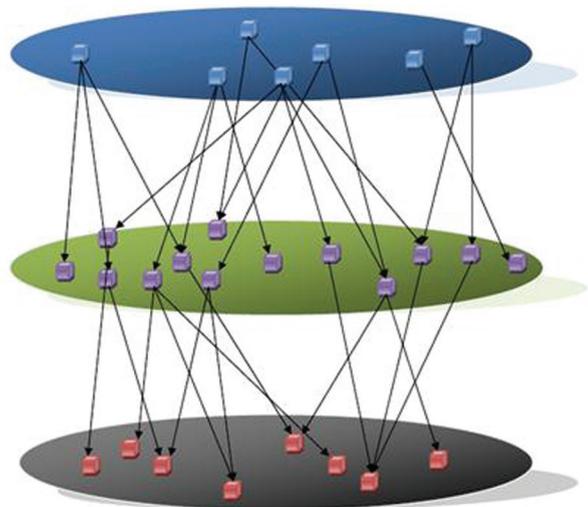


图1 语义网络立体空间模型

或几种. 同时, 为了减少语义模糊性, 计算节点相似度后, 将相似度高的节点进行合并. 使得语义查询和社区发现可以快速转换到合适的语义平面上, 从而可以提高搜索的速度和查全率.

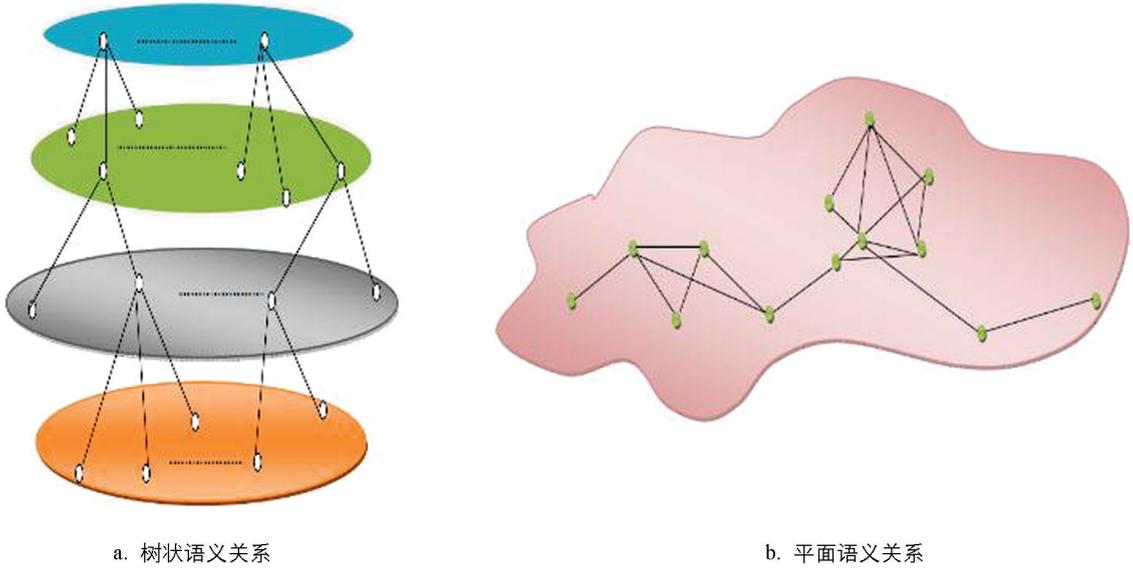


图 2 语义关系拓扑结构

2.2 模型描述

本文根据语义链的特性构造空间立体语义链网络模型.

2.2.1 相等、引用与序列

在平面上, 用有向图表示各节点间的关系. 定义空间模型中, 第 n 个平面上的图为 $G_n = \langle V_n, Equ_n, Ref_n, Seq_n \rangle$, 其中: V_n 是第 n 个平面上所有点的集合, Equ_n 代表第 n 个平面图中所有语义链类型为“相等”的节点集合, Ref_n 与 Seq_n 同理.

若多个节点间的语义链是相等的, 则合并所有的有此语义链关系的点为一点. 如图 3 网络中节点 A, B, C 间有“相等”语义链, 则合并为 A (一般按第一个节点进行合并), 节点 D 与 B 间语义链为其他类型, 则保留, 最后合并为两个节点 A 和 D. “相等”语义链及合并结果见图 3.



图 3 “相等”语义链及合并结果

假设两节点 A, B 的语义相似度为 α , 当 $\alpha \geq 0.5$ 时, 将此语义链的类型归入相等, 若 $\alpha < 0.5$ 时, 则认为 A, B 语义不相关.

Ref 为图中所有语义链类型为“引用”的节点关系集合, $Ref_n = \langle r_{n1}, r_{n2} \cdots r_{nm} \rangle$, r_{ni} 为第 n 个平面上第 i 条“引用”语义链. Seq 是所有语义链类型为“序列”的节点集合, $Seq_n = \langle s_{n1}, s_{n2} \cdots s_{nm} \rangle$, s_{ni} 为第 i 个序列. “引用”语义链和“序列”语义链的表示见图 4.

“引用”语义链与“序列”语义链属于不同类型. “引用”语义链具有很强的时序性, 时间上, 要求前面的节点不能引用后面的节点.

2.2.2 蕴涵与子类

每个以树为单位的纵向搜索是从根节点语义 (Root Semantics) 开始, 以蕴涵或子类语义链作为搜索路径连接起来的各点. 路径上的节点具有一定的有序性, 且可以通过一个节点查找到其他节点. 蕴涵或子类可以表示为这样的树的集合. 纵面上的树结构可以表示为语义节点、“子类”语义链、“蕴涵”语义链的集合, $T = \langle V_n, Sub, Imp \rangle$. “子类”语义链和“蕴涵”语义链的表示见图 5.



图 4 “引用”语义链和“序列”语义链

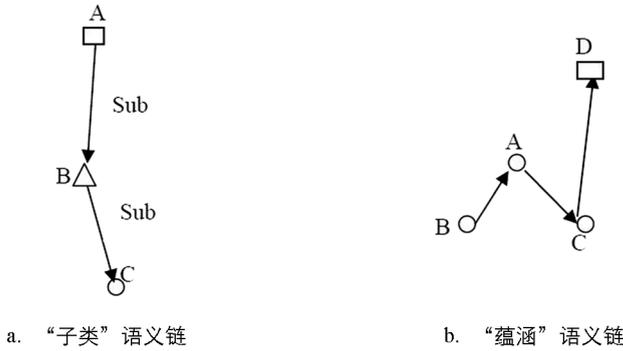


图 5 “子类”语义链和“蕴涵”语义链

“子类”语义链表示为 $Sub = \cap \langle N_i, \dots, N_n \rangle$, 其中 $\langle N_i, \dots, N_n \rangle$ 是由节点 N_i 到 N_n 的一条“子类”语义链. 按定义, “子类”语义链有传递关系. 而“蕴涵”通常情况下并不单纯指简单的包含关系. 我们重点关注语义链关系, 即在某条或某几条语义链存在的情况下, 是否有某种特殊的且研究者感兴趣的语义链产生.

根据节点对象间相似度的度量和修正需要, 结合文献[11]中语义链关系定义及本文中社区发现定义, 提出语义推理表(表 1).

表 1 语义链接网络的推理规则

	推理规则	注释
1	$P_i - Equ \rightarrow P_i$	—
2	$P_i - Equ \rightarrow P_j$	$P_j - Equ \rightarrow P_j$
3	$P_i - Equ \rightarrow P_j, P_j - Equ \rightarrow P_k$	$P_i - Equ \rightarrow P_k$
4	$P_i - Equ \rightarrow P_j, P_j - Sim \rightarrow P_k$	$P_i - Sim \rightarrow P_k$
5	$P_i - Imp \rightarrow P_j, P_j - Imp \rightarrow P_k$	$P_j - Imp \rightarrow P_k$
6	$P_i - Sub \rightarrow P_j, P_j - Sub \rightarrow P_k$	$P_j - Sub \rightarrow P_k$
7	$P_i - Imp \rightarrow P_j, P_j - Sub \rightarrow P_k$	$P_j - Imp \rightarrow P_k$
8	$P_i - Imp \rightarrow P_j, P_j - Ref \rightarrow P_k$	$P_j - Ref \rightarrow P_k$
9	$P_i - Sub \rightarrow P_j, P_j - Imp \rightarrow P_k$	$P_j - Imp \rightarrow P_k$
10	$P_i - Sub \rightarrow P_j, P_j - Ref \rightarrow P_k$	$P_j - Ref \rightarrow P_k$
11	$P_i - Non \rightarrow P_j, P_j - Sim \rightarrow P_k$	$P_j - Non \rightarrow P_k$
12	$P_i - \emptyset \rightarrow P_j, P_j - Sim \rightarrow P_k$	$P_j - Non \rightarrow P_k$

其中: $Imp = \cap \langle N_i, \dots, N_n \rangle$ 表示可能产生推理的各个语义节点集合, Equ 代表相等关系, Sim 代表相似关系, Ref 代表引用关系, Sub 代表子类关系, Non 代表无关系.

2.2.3 相似度计算

用一个词空间的向量 $Vector(e)$ 表示本文中的一个实体 e , 每个维度对应一个词, 维度大小取值表明了

这个词在刻画 e 时的相对重要性. 用词空间中的向量 $Vector(q)$ 表示一条基于关键词的查询 q . e 和 q 的相似程度可以被描述为 $Vector(e)$ 和 $Vector(q)$ 的夹角余弦值.

语义结构的相似度取决于两个基本的要素: ① 语义节点组成社区的叶子结点, 且查询也是由叶子结点构成, 所以与社区的语义结构相关; ② 节点的祖先节点的相似程度.

表 2 给出了实现相似度算法所调用函数及其注释.

表 2 函数及其注释

函数	注释
Peer (N_i)	语义节点 N_i 的语义映射节点
Root(N_i)	语义节点 N_i 的根节点
Length (N_i, N_j)	语义节点 N_i 到 N_j 路径上节点的个数
Max-Semantic-Clique(N_i, N_j)	包括语义节点 N_i 与 N_j 的最大语义群
Min-Common-Sub-Tree(N_i, N_j)	包括语义节点 N_i 的最小公共子树
Semantic-Node-Similarity-Degrees (N_i, N_j)	语义节点 N_i 与 N_j 之间的相似度

N_i 与 N_j 语义结构相似度计算如下:

算法 1 Semantic-Structure-Similarity-Degrees (N_i, N_j)

IF N_i 属于社区中某一个最大语义群

THEN

$T = \text{Max-Semantic-Clique}(N_i, N_j)$

ELSE

$T = \text{Min-Common-Sub-Tree}(N_i, N_j)$

END IF

RootSet $N_i = T$

NodeSet = $\{N_i, \dots, \text{Root}(N_i)\} \cup \text{RootSet } N_i$

IF Length (N_i, N_j) = 1

THEN

Semantic-Structure-Similarity-Degrees (N_i, N_j) = Semantic-Node-Similarity-Degrees (N_i, N_j)

Return()

ELSE

$\vec{FV} = \text{Semantic-NodeSet-Similarity-Degrees}(\text{NodeSet}, N_j)$

FOR N_k IN NodeSet

IF $N_k = N_j$

THEN $W_{N_{kj}} = \frac{1}{2}$

ELSE IF N_k IN RootSet N_j

THE $N_k = \text{length}(N_k, \text{RootSet } N_j)$; $W_k = 1 - \sum_{l=1}^{n-1} \tau_{N_l} = \frac{1}{2}^{n-1}$;

ELSE IF $N_k \neq N_j$

THE $N_k = \text{length}(N_i, N_k)$; $W_{N_j} = \frac{1}{2}^k$;

$$\text{Semantic-Structure-Similarity-Degrees}(N_i, N_j) = \frac{\vec{W} \cdot \vec{FV}}{|\vec{W}| |\vec{FV}|}$$

END IF

算法 1 首先提取节点在网络中的结构, 如果节点互为邻居节点, 那么他们的结构相似度就是他们的语义相似度. 如果节点不为邻居节点, 则找出指向该节点的上层节点, 然后建立相似度向量 \vec{FV} , \vec{FV} 包含从根节点到节点 N_j 与 N_i 的层次节点的相似度向量 $[Vec_i, \dots, Vec_k, \dots, Vec_{Root(N_j)}]$, 其中:

$$Vec_i = \begin{cases} 0 & \text{Semantic-Node-Similarity-Degrees}(N_i, N_j) \\ \text{对应 } N_k \text{ 的映射节点不属于 } N_k \text{ 的映射路径} & \text{其他} \end{cases}$$

得到节点的相似度向量后, 还需要得到节点的权重向量, W_{N_k} 则表示节点 N_i 与节点之间的重要程度.

$$W_{N_k} = \begin{cases} \frac{1}{2} & N_k = N_j \\ \frac{1}{2^k} & k = \text{length}(N_i, N_k) \text{ and } N_k \neq N_j \\ 1 - \sum_{l=1}^{n-1} \omega_{N_L} = \frac{1}{2^{n-1}} & n = \text{length}(N_k, N_j) \text{ and } N_k = \text{Root}(N_j) \end{cases}$$

$\vec{\omega}$ 是权值向量 $[W_{N_i}, \dots, W_{N_k}, \dots, W_{Root(N_i)}]$, 表示节点在另一个节点结构中的重要性, 通过权值向量和特征向量的结合得到了结构的语义相似度.

算法 2 基于算法 1, 在用户自定义权重 W 的影响下, 输出集合 Set_{n_m} 中的各个节点与 N_i 节点的语义结构相似度组成的向量.

算法 2 Semantic-Structure-Similarity-Degrees-extend (N_i , NodeSet)

NodeSet_{*mm*} = { N_n, \dots, N_m }

FOR (N_j in NodeSet):

Similarity-Degrees_{*N_j*} = Semantic-Structure-Similarity-Degrees(N_i, N_j)

Semantic-Structure $\vec{\text{Similarity-Degrees}}$ = {Similarity-Degrees_{*N_n*}, \dots , Similarity-Degrees_{*N_m*}}

$$\text{Semantic-Structure-Similarity-Degrees}(P_i, P_j) = \frac{\vec{W} \cdot \text{Semantic-Structure}\vec{\text{Similarity-Degrees}}}{|\vec{W}| |\text{Semantic-Structure}\vec{\text{Similarity-Degrees}}|}$$

其中, \vec{W} 是用户定义的语义节点重要程度.

Semantic-Structure $\vec{\text{Similarity-Degrees}}$ 代表的是集合 NodeSet_{*mm*} 中的各个节点与 N_i 节点的语义结构相似度组成的向量.

2.2.4 语义链网络立体空间构建

算法 3 描述语义链网络空间模型的构建.

算法 3 构建语义链网络立体空间模型(DataSet)

Filter(DataSet); //网页抓取的数据通过本体集合过滤

Set_data = Init(DataSet); //初始化输入集合 Set_data

$T, G = \text{Trans}(Set_data)$ //采用本体之间的关系遍历整个社区集, 确定一部分节点的 $V_n, Equ, Ref, Seq, Sub, Imp$, 使用推理表格遍历已构造的图谱, 加速节点间关系的收敛.

$M = \text{Construct}(G, T)$ //将用户的 ID 作为主体主键, 通过 Semantic-Structure-Similarity-Degrees-extend 算法构造空间立体模型 $M = (G, T)$, 其中 $G = \langle V_n, Eqa, Ref, Seq \rangle$, $T = (V_n, Sub, Imp)$;

NodeSim = Norm($M.SimVec$) //归一化各节点语义结构相似度

Sort (NodeSim { $N_{Sim_1}, \dots, N_{Sim_n}$ }) //对其进行排序

Mark = 0 //标记已被计算的节点(矩阵)为 0

IF $N_{Sim_i} = 1$

```

THEN FOR  $N_i$  in  $G.EqaSet$  or  $T.SubSet \in N_{Sim_i}$ 
   $N_i \subset OutPut$ 
   $Mark[i] = 1$ 
ELSEIF  $N_{Sim_i} < 0.2$ 
  THEN FOR  $N_i$  in  $T.SubSet$  or  $G.EqaSet \in N_{Sim_i}$ 
     $N_i \not\subset OutPut$ 
     $Mark[i] = 1$ 
  FOR  $Mark[i] = 0$  and  $Sim_{N_i} > 0.5$  //节点间关联度对相似度的修正
     $Important_{N_i} = 1$ 
     $Important_{N_j} = 0.8$ 
  FOR  $N_j$  in  $ImpSet$ 
     $Important_{N_j} = Important_{N_j} + Sim_{N_i} * (1 - \alpha) Important_{N_i}$ 
  FOR  $N_j$  in  $SeqSet$ 
     $Important_{N_j} = Important_{N_j} + Sim_{N_i} * (1 - \beta) Important_{N_i}$ 
  FOR  $N_j$  in  $RefSet$ 
     $Important_{N_j} = Important_{N_j} + Sim_{N_i} * (1 - \gamma) Important_{N_i}$ 
  FOR  $Mark[j] = 0$ 
     $Sim_{N_j} = Important * Sim_{N_j}$ 
    IF  $Sim_{N_j} > 0.5$ 
       $N_j \subset OutPut$ 
       $Mark[j] = 1$ 
Return qutPut

```

3 实验及分析

在本实验中,我们通过模拟实验验证社区发现算法的准确性.通过选取 Facebook 中的样本随机建立二层社交网络.通过语义推理得到稠密矩阵,并转化为无向图,再对无向图进行社区划分证明本算法的有效性.然后,选取主流的社区发现算法 K-means 和隐语义模型(latent factor model, LFM)作为本模型算法的对比算法,进行对比实验.结果发现, K-means 的社区划分准确性比其他两个算法更低, LFM 和本文算法的社区划分准确度近似,但当社区大小增长到 1 000 过后,本文算法的模块度逐渐显示出优势.模块度是衡量社区划分的强度指标,以模块度值的大小来评价社区划分的优劣,模块度值越接近 1,表示对社区结构的划分越好.模块度值越大,社区内部连接越紧密,社区间连接越稀疏,划分效果越好^[1].

实验采用斯坦福大学提供的 ego-Facebook 的数据集. ego-Facebook 数据集是从 App 端采集的 Facebook 用户的数据,包含了用户的属性、社交圈(Circles)和 ego network,数据已被做了脱敏处理.数据共有 4 039 个用户和 88 234 条连边.其社交网络图的密度是 0.005 409 92,图直径为 17,平均距离为 4.337 746,能够很好代表现实世界中的好友关系.

本文实验在 CPU 为 AMD Ryzen 7 5800Hz,内存为 16GB,系统为 Win10 的电脑上运行.

3.1 社区发现模型的实验结果和分析

3.1.1 社区发现模型有效性

为了证明本文提出的社区发现模型的有效性,本文先随机选取 ego-Facebook 中的一个用户,建立两层社交网络关系图,图中包含 252 个节与 4875 条边.本文构造原始的不带权有向图,推理得到稠密矩阵,此时的边数为 10 376,包含了所有的语义关系.本文将稠密矩阵作为相似度计算的输入得到无向图.本文对

无向图中的社区节点进行划分得到社区发现结果. 图 6 给出了原始的 Facebook 好友有向图、转化的 Facebook 好友有向图、Facebook 好友无向图、Facebook 好友社区发现结果.

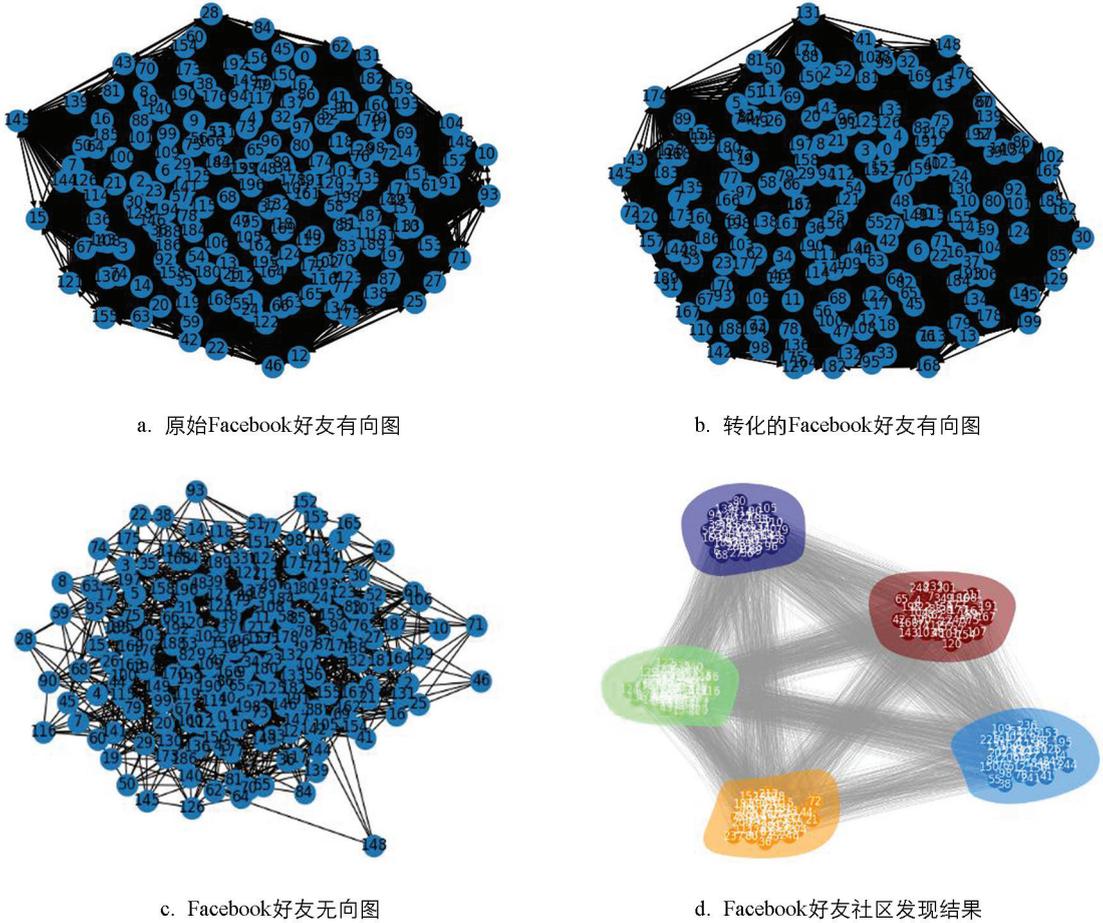


图 6 基于 Facebook 的社区发现步骤及结果

3.1.2 算法性能对比

我们选取社区发现主流的 k -means 算法和隐语义模型作为本模型算法的对比算法. 实验中, 从 0 个节点开始, 每次从数据库中随机取出 40 个新的 Facebook 用户节点添加到原有的网络中. 通过模拟发现, 3 个算法在拟合下都是线性增长. 随着网络节点的增加, k -means 算法对社区的划分结果不稳定, 呈现大幅波动状态. 3 种算法对比结果见图 7.

LFM 和本文结果在社区划分结果方面差不多, 但当社区大小增长到 1 000 过后, 本文算法的模块化程度逐渐显示出优势.

稳定性方面, 如图 8, 本文算法模块化方差为 $5.267\ 942e-06$ 低于 LFM 算法模块化方差 $5.814\ 608e-06$, 由此可见本文的算法更加稳定.

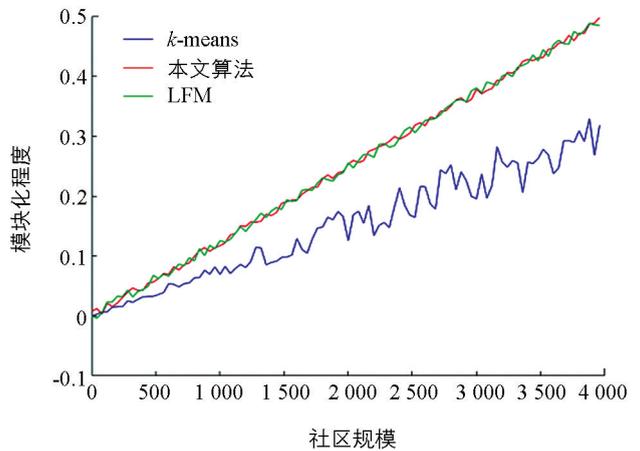


图 7 算法对比

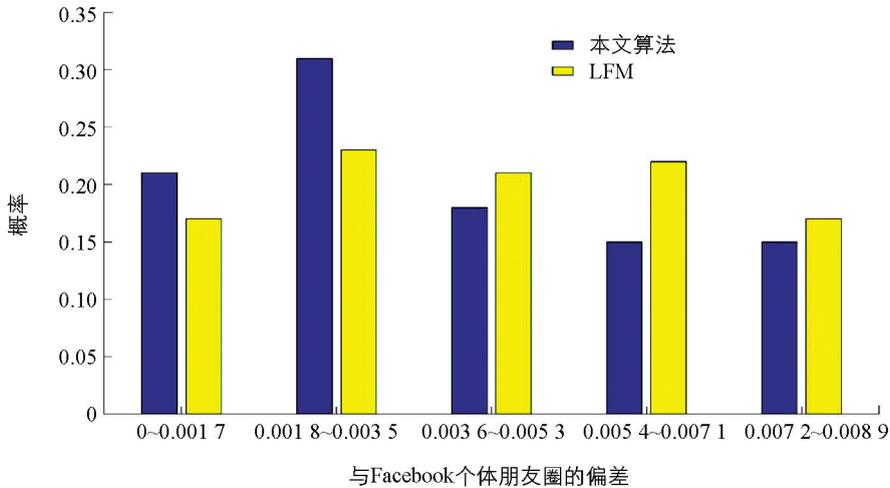


图 8 本文算法与 LFM 算法个人 circle 结果比较

4 结论

本文提出基于语义推理的网络社区发现模型,创新地从语义网络的角度,探讨网络社区的构成及分割.模型基于语义的平面和纵面两种特性,利用图加树的空间拓扑结构,进行基于语义推理的社区发现.同时,本模型通过语义本身特性,将基于语义链的搜索简化到了层次范畴.实验结果显示,在模块度的评价指标下,本文提出的模型对社区划分结果优于 LFM 和 k -means 算法.另一方面,由于语义的分割及关系的复杂性会影响本模型速度,期望在下一步的研究中解决该问题.

参考文献:

- [1] CLAUSET A, NEWMAN ME J, MOORE C. Finding Community Structure in very Large Networks [J]. Physical Review E, Statistical, Nonlinear, and Soft Matter Physics, 2004, 70(6): 066111.
- [2] LI W M, ZHOU X K, YANG C, et al. Multi-Objective Optimization Algorithm Based on Characteristics Fusion of Dynamic Social Networks for Community Discovery [J]. Information Fusion, 2022, 79: 110-123.
- [3] NAIK D, RAMESH D, GANDOMI A H, et al. Parallel and Distributed Paradigms for Community Detection in Social Networks: a Methodological Review [J]. Expert Systems With Applications, 2022, 187: 115956.
- [4] 端祥宇, 袁冠, 孟凡荣. 动态社区发现方法研究综述 [J]. 计算机科学与探索, 2021, 15(4): 612-630.
- [5] YANG Z L, ZHANG W J, YUAN F, et al. Measuring Topic Network Centrality for Identifying Technology and Technological Development in Online Communities [J]. Technological Forecasting and Social Change, 2021, 167: 120673.
- [6] 阎海玲. 复杂网络社区发现方法综述 [J]. 信息与电脑(理论版), 2018(20): 162-163, 169.
- [7] 陈洒然. 面向社会网络的网络抽样与统计推断方法研究 [D]. 长沙: 国防科技大学, 2018.
- [8] SATULURI V, PARTHASARATHY S. Symmetrizations for Clustering Directed Graphs [C] //EDBT/ICDT'11: Proceedings of the 14th International Conference on Extending Database Technology. 2011: 343-354.
- [9] 张海燕, 梁循, 周小平. 针对有向图的局部扩展的重叠社区发现算法 [J]. 数据采集与处理, 2015, 30(3): 683-693.
- [10] BERNERS-LEE T, HENDLER J, LASSILA O. The Semantic Web [J]. Scientific American, 2001, 284(5): 34-43.
- [11] 谭茨, 张进, 夏立新. 语义网络发展历程与现状研究 [J]. 图书情报知识, 2019(6): 102-110.
- [12] ANTONIOU G, HARMELEN F. 语义网基础教程 [M]. 陈小平, 译. 北京: 机械工业出版社, 2008.
- [13] 黄恒琪, 于娟, 廖晓, 等. 知识图谱研究综述 [J]. 计算机系统应用, 2019, 28(6): 1-12.
- [14] 庞美玉. 基于 TAP 的上下文知晓框架研究 [D]. 太原: 太原理工大学, 2006.