软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software, 2022,33(8):3103−3114 [doi: 10.13328/j.cnki.jos.006283]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于感染结果的传播网络推断方法*

赛影辉[1], 王明鑫[1], 陈 畅[1], 雷伯涵[2], 侯叶俏[1], 李翔翔[3], 孙月明[1], 陈 旭[1]

[1](武汉大学 计算机学院, 湖北 武汉 430072)

[2](浙江大学 计算机科学与技术学院, 浙江 杭州 310027)

[3](航天恒星科技有限公司, 北京 100086)

通信作者: 陈旭, E-mail: xuchen@whu.edu.cn

摘 要: 为揭示传播网络中节点之间的父子影响关系, 现有工作大多需要知道节点的感染时间, 而该信息往往只有通过对传播过程进行实时监控才能获得. 研究如何基于传播结果来学习获得传播网络中节点之间的父子影响关系. 传播结果只包含每个传播过程中节点的最终感染状态, 而节点的最终感染状态在实际中往往比节点的感染时间更容易获得. 提出了一种基于条件熵的方法来推断网络中每个节点的潜在候选父节点. 此外, 能够通过从基于条件熵的推断结果中发现并修剪那些实际不太可能存在的父子影响关系来优化最终的影响关系推断结果. 在人工网络和真实网络上的大量实验, 验证了该方法的有效性和运行效率.

关键词: 传播网络推断; 影响关系; 感染结果

中图法分类号: TP309

中文引用格式: 赛影辉, 王明鑫, 陈畅, 雷伯涵, 侯叶俏, 李翔翔, 孙月明, 陈旭. 基于感染结果的传播网络推断方法. 软件学报, 2022, 33(8): 3103–3114. http://www.jos.org.cn/1000-9825/6283.htm

英文引用格式: Sai YH, Wang MX, Chen C, Lei BH, Hou YQ, Li XX, Sun YM, Chen X. Diffusion Network Inference Based on Infection Results. Ruan Jian Xue Bao/Journal of Software, 2022, 33(8): 3103–3114 (in Chinese). http://www.jos.org.cn/1000-9825/6283.htm

Diffusion Network Inference Based on Infection Results

SAI Ying-Hui[1], WANG Ming-Xin[1], CHEN Chang[1], LEI Bo-Han[2], HOU Ye-Qiao[1], LI Xiang-Xiang[3], SUN Yue-Ming[1], CHEN Xu[1]

[1](School of Computer Science, Wuhan University, Wuhan 430072, China)

[2](College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China)

[3](Space Star Technology Co. Ltd., Beijing 100086, China)

Abstract: To reveal parent-child influence relationships between nodes in a diffusion network, most prior work requires knowledge of node infection time, which is possible only by carefully monitoring the diffusion process. This work investigates how to solve this problem by learning from diffusion results, which contain only the final infection statuses of nodes in each diffusion process and are often more easily accessible in practice. A conditional entropy-based method is presented to infer potential candidate parent nodes for each node in the network. Furthermore, the inference results are able to be refined by identifying and pruning the inferred influence relations that are unlikely to exist in reality. Experimental results on both synthetic and real-world networks verify the effectiveness and efficiency of our approach.

Key words: diffusion network inference; influence relationships; infection results

了解传播网络中节点之间的影响关系, 对于理解传播网络的内在属性、设计有效的策略来促进或防止网络中的传播事件的发生有着重要的意义. 但对于许多真实的传播网络, 如疾病传播网络, 其中的节点(即人类

* 基金项目: 民用航天“十三五”技术预先研究项目(B0301); 湖北省技术创新专项重大项目(2017AAA125); 武汉市应用基础前沿项目(2018010401011288)

收稿时间: 2019-09-27; 修改时间: 2020-01-23, 2020-09-09; 采用时间: 2020-12-12; jos 在线出版时间: 2021-08-03

个体)之间的影响关系往往是不能直接观察到的, 因此常常需要基于历史传播数据来推断这些影响关系. 这类问题通常被称为传播网络推断, 在社交网络[1]、信息传播[2]、传染病防控[3]和病毒式营销[4]等领域受到广泛的关注.

为进行传播网络推断, 现有方法大都采用如下假设: 一个节点的感染有很高概率是由在它之前不久被感染的节点引起的[5]. 因此, 这些现有方法需要知道节点感染发生的时间. 换言之, 如要使用这些现有方法, 用户需要对每个传播过程进行监测, 准确记录各节点感染的发生时间. 然而在现实生活中, 对传播过程进行全程监控往往费时费力, 甚至很多时候是不可行的, 特别是当传播过程持续时间较长、节点空间分布较广、监控需要人力和资源较多时. 此外, 一些不可避免的客观因素, 如各个节点不确定的潜伏期(即从感染发生到出现可观测感染症状的时间), 也会使得节点的感染时间难以确定[6−9].

目前, 仅有极少量方法提出了在没有感染时间信息的情况下重建传播网络[10,11], 包括:

(1) 从路径轨迹学习(以下简称 PATH);

(2) 从感染节点的初始集和结果集学习(以下简称 I2R).

PATH 算法需要知道所有节点子集,每个子集由 3 个节点组成, 这些节点在传播路径中连接, 尽管精确的传播路径通常是很难跟踪的. 而 I2R 算法需要事先知道感染源(最初被感染的节点)以及目标网络中影响关系数量等先验知识, 即使这些先验知识在实践中是很难获得的.

为避免现有传播网络推断方法的局限性, 本文研究如何仅仅基于传播结果, 即各节点在各传播过程中的最终感染状态, 来完成传播网络推断. 与现有方法所需的感染时间等信息相比, 节点的最终感染状态在大多数情况下更容易获得. 为达到以上研究目标, 我们提出了 LDR(learning from diffusion results). 该算法通过为每个节点选择一组最有可能的潜在父节点来揭示潜在的节点间影响关系. 为此, LDR 算法采用了一种基于条件熵的感染结果概率生成模型, 用来度量所观测到的节点感染状态结果是由某种潜在父节点集合所生成的概率. 较低的条件熵表示较高的生成概率. 同时, LDR 算法采用一种改进的互信息保障推断出的潜在父节点与其子节点之间的影响关系强度不至于过低. 在使用条件熵和改进互信息为各节点推断出最有可能的、具有明显影响关系的潜在候选父节点集合后, 需要进一步判断哪些潜在父节点实际并不存在, 而是由于其他潜在父节点的中介作用而被引入潜在父节点集合的. 为此, LDR 算法采用一种改进的条件互信息来排除其他潜在父节点的中介作用, 衡量各个潜在父节点的感染是否与其子节点的感染具有直接相关性, 并将直接相关性极低的潜在父节点从最终的推断结果中去除.

本文第 1 节回顾传播网络推断的现有相关工作. 第 2 节给出研究问题的问题描述. 第 3 节介绍本文所提出的 LDR 算法. 第 4 节报告实验结果. 第 5 节对全文做出总结.

1 相关工作

现有的传播网络推断方法可以被划分为两大类.

(1) 基于时间信息的传播网络推断方法;

(2) 不依赖时间信息的传播网络推断方法.

1.1 基于时间信息的传播网络推断方法

现有绝大多数传播网络推断方法都需要节点感染的确切时间信息. 通常, 这些方法将按先后顺利记录的节点感染时间称之为级联(cascades)数据. 目前, 使用级联数据的方法大致可分为以下 3 类, 即:

(1) 基于凸优化的方法;

(2) 基于子模性质的方法;

(3) 基于嵌入空间的方法.

基于凸优化的方法将传播网络推断问题转化为凸优化问题, 旨在找出最有可能产生给定级联数据的潜在传播网络. 为此, 此类方法使用连续二次最优化[12]、EM 算法[13]、块坐标下降[14]、随机和近似梯度法[15,16]、生存论[2]、稀疏恢复[17]、解耦成多个多可并行问题[18]等技术手段求解对应的凸优化问题. 通常, 这类方法在

树状或稀疏网络上可获得较好的推断结果.

基于子模性质的方法将传播网络推断问题转化为子模优化问题, 因为它们所构造的级联的似然函数具有子模性质. NetInf[19]和 MulTree[20]是这类方法中两种代表性算法. 由于目标函数的子模性, 上述两种算法均采用贪心算法获得近似最优解. 其主要区别在于: 在优化过程中, NetInf 算法只考虑最可能的传播树来追求更高的效率, 而 MulTree 算法则考虑每个级联所支持的所有传播树来获得更好的准确性.

基于嵌入空间的方法尝试将节点映射到一个潜在的嵌入空间, 其中, 两个节点之间的距离表示传播概率(或传输速率). 这类方法常假设传播概率服从韦布尔分布[21]或均匀分布[1,22], 或者采用核函数建模传播概率[23],然后根据观测到的级联数据学习两两节点间的传播概率. 虽然此类方法没有显式地揭示传播网络结构, 但是它们使用户能够通过低维可视化空间直观观察节点之间的影响关系.

上述 3 种类型的传播网络推断方法都需要完整而正确的级联数据. Abrahao 等人的研究表明: 在给予足够的完整正确的级联数据时, 即使使用一些简单的图重构方法, 也可以比较准确地推断出目标传播网络[24]. 然而, 在现实中观测到的级联数据常可能包含部分不正确的感染时间, 或者有时会缺少部分时间上的观测结果. 因此, 也有部分研究旨在减轻不正确和缺失的时间信息的影响[25−27]. 此外, Shaghaghian 等人研究了当级联数据记录的不是节点感染时间而是节点感染症状出现时间的情况下, 如何推断感染时间和目标传播网络[28,29]. 这些研究都可视为上述 3 种类型方法的合理补充, 但依然属于基于时间信息的传播网络推断方法.

与基于时间信息的方法相比, 本文提出的 LDR 算法只需要节点的最终感染状态. 相比时间信息, 节点最终感染状态往往更易获得. 因此, 本文方法具有更好的适用性, 并可以天然地避免不正确和缺失时间信息带来的问题.

1.2 不依赖时间信息的传播网络推断方法

到目前为止, 只有少量的不依赖时间信息的方法被提出来用于重建传播网络. 这些方法或者需要知道传播路径(PATH), 或者需要知道初始的和最终的感染节点集合(I2R).

- PATH 算法将由传播路径连接的感染节点三元组作为输入, 其中, 每个三元组由 3 个节点组成, 它们沿着网络中的传播路径被先后感染[11]. 虽然 PATH 算法具有不少优点, 例如坚实的数学基础和较低的计算成本, 但它需要沿传播路径连接的感染节点三元组. 由于传播路径常常不可观测, 要获得沿传播路径连接的感染节点三元组是非常困难的. 即使用户具有足够的完整正确的级联数据, 要推断出完整正确的沿传播路径连接的感染节点三元组仍然是非常具有挑战性的.
- I2R 算法研究了如何利用初始的和最终的节点感染状态来进行传播网络推断的问题[10]. 为此, I2R 计算了每个节点 u 对另一个节点 v 的提升效果, 来衡量在已感染 u 的情况下, v 感染概率的提升. I2R 通过不断找到当前提升效果最大的一对节点来发现潜在的节点间影响关系, 并在它们之间加入一条有向边来表示该潜在影响关系. 当目标传播网络中的有向边数未知时, I2R 将会不断地添加有向边, 直到所有节点都相连为止.

与上述的不依赖时间信息的方法相比, 本文提出的 LDR 算法只需要网络节点的最终感染状态, 不再需要其他任何关于感染路径的信息或传播网络的先验知识. 因此, LDR 算法在实际应用中更加广泛适用范围.

2 问题描述

一个传播网络可表示为有向图 $G=(V,E)$, 其中, $V=\{v_1,v_2,\ldots,v_n\}$为网络中 n 个节点的集合, E 为节点之间 m 条有向边的集合(即影响关系集合). 从父节点 v_i 到子节点 v_j 的边表示当 v_i 被感染、v_j 未被感染时, v_i 将以一定的传播概率感染 v_j(可视为边的权值). 由于部分现有研究成果已经解决在已知传播网络拓扑结构的情况下如何计算各条边上的传播概率[30], 本文中, 我们主要关注如何利用节点感染状态结果来推断得到传播网络拓扑结构. 基于感染状态结果的传播网络拓扑结构推断问题可以描述如下[31].

- 给定: 集合 $S=\{s^1,\ldots,s^\beta\}$是传播网络 G 在β次相互独立的历史传播过程结束时的节点感染状态结果, 其

中，$S^{\ell}=(x_1^{\ell},...,x_n^{\ell})$ 是一个记录各节点最终感染状态的 n 维向量，$x_i^{\ell}\in\{0,1\}$ (0 为未感染的状态和 1 为感染状态)为第 ℓ 次传播过程结束时节点 $v_i\in V$ 的感染状态($\ell\in\{1,...,\beta\}$);

- 推断: 传播网络 G 的边集 E.

3 LDR 算法

在本节中，首先介绍基于条件熵的概率生成模型来衡量节点感染状态结果是由某种潜在父节点集合所生成的概率；然后，我们讲解如何通过为每个节点找出一组最有可能的潜在父节点集合来推断潜在的影响关系；最后，展示如何进一步推断出的潜在父节点集合，从中识别和删除可能实际不存在的潜在父节点，并将对本文所提方法的时间复杂性进行分析.

3.1 概率生成模型

要使用阶段感染状态结果来推断传播网络的拓扑结构，最主要的工作是提出节点感染状态结果的概率生成模型，即以传播网络中各个节点的感染状态 $X_1,\ldots,X_n$ 为变量的联合概率分布 $p(X_1,\ldots,X_n)$. 因为在传播过程中，各节点仅能够被其父节点感染，所以我们可以将联合概率分布 $p(X_1,\ldots,X_n)$以如下方式重新定义:

$$p(X_1,...,X_n)=\prod_{i=1}^{n}p(X_i\mid X_{F_i})p(X_1,...,X_n) \tag{1}$$

其中，F_i 为节点 $v_i\in V$ 的父节点，X_{F_i} 为节点 v_i 的父节点 F_i 的感染状态变量. 令有向图 $G'=(V,E')$为所要推断的目标传播网络 G 的一个估计. 在给定 G'的条件下，基于公式(1)，传播网络中各节点的感染状态观测数据 $S=\{S^1,\ldots,S^{\beta}\}$的对数似然度函数 $\log L(S|G')$为

$$\left.\begin{aligned}\log L(S\mid G')&=\log\left(\prod_{\ell=1}^{\beta}p(X_1=x_1^{\ell},...,X_n=x_n^{\ell})\right)\\&=\log\left(\prod_{\ell=1}^{\beta}\prod_{i=1}^{n}p(X_i=x_i^{\ell}\mid X_{F_i'}=\pi_i^{\ell})\right)\\&=\sum_{\ell=1}^{\beta}\sum_{i=1}^{n}\log(p(X_i=x_i^{\ell}\mid X_{F_i'}=\pi_i^{\ell}))\end{aligned}\right\} \tag{2}$$

其中，F_i' 是节点 v_i 在传播网络 G'中的父节点集合；π_i^{ℓ} 是在第 ℓ 次传播过程结束时，节点 v_i 的父节点集合 F_i' 中的各父节点的最终感染状态. $\log L(S|G')$代表由传播网络 G'生成各节点最终感染状态数据 S 的概率的对数值. $\log L(S|G')$的数值越大，则表示当前的传播网络 G'是对目标传播网络 G 的一个更好的估计. 令 N_{x_i,π_i} 为变量 X_i 和变量 $X_{F_i'}$ 在节点感染状态观测数据 S 中不同取值的统计量，即在 β 次传播过程结束后，节点 v_i 及其父节点 F_i' 的感染状态所构成的数值对 (x_i,π_i)的不同取值情况在观测数据 S 出现的次数；令 N_{π_i} 为父节点 F_i' 的感染状态对应的变量 $X_{F_i'}$ 在节点感染状态观测数据 S 中不同取值的统计量. 显然，$N_{\pi_i}=\sum_{x_i\in\{0,1\}}N_{x_i,\pi_i}$ 恒成立. 那么，对数似然度函数 $\log L(S|G')$可以改写为如下形式:

$$\log L(S\mid G')=\sum_{i=1}^{n}\beta\sum_{x_i,\pi_i}\frac{N_{x_i,\pi_i}}{\beta}\log\frac{N_{x_i,\pi_i}}{N_{\pi_i}}=\sum_{i=1}^{n}\beta(-H(X_i\mid X_{F_i'})) \tag{3}$$

其中，$H(X_i\mid X_{F_i'})$ 为在已知变量 $X_{F_i'}$ 的条件下，变量 X_i 的条件熵. 具体计算方法如下:

$$H(X_i\mid X_{F_i'})=-\sum_{x_i,\pi_i}\frac{N_{x_i,\pi_i}}{\beta}\log\frac{N_{x_i,\pi_i}}{N_{\pi_i}} \tag{4}$$

条件熵 $H(X_i\mid X_{F_i'})$ 的值越小，则表示由传播网络 G'产生节点 v_i 最终感染状态的可能性越大. 由以上公式(3)可知: 若要得到能够最大化对数似然度函数 $\log L(S|G')$的传播网络拓扑结构最优解，只需要从节点集合 V 中为每个节点 v_i 找到其父节点集合 F_i'，并且这个父节点集合能够使得条件熵 $H(X_i\mid X_{F_i'})$ 的值最小化. 然而，该方法会使得得到的传播网络拓扑结构复杂度过高. 这是因为若节点 v_i 的父节点集合 F_i' 越大，即其父节点的数

量增多, 在多余的变量的影响下, 条件熵的值会减小[32]. 此外, 根据独立级联传播模型(IC 模型常应用于传播网络推断和影响最大化问题)[12,33]的定义, 每个节点只能被一个已感染的父节点所感染, 而不是被多个已感染的父节点所共同感染. 因此在计算节点 $v_i \in V$ 的条件熵 $H(X_i \mid X_{F_i'})$ 时, 每次仅仅考虑潜在父节点集合中的一个节点, 即对于节点 v_i, 只计算 $H(x_i|x_j)$的值, 其中, $i \neq j$. 若 $H(x_i|x_j)$的值最小, 则节点 $v_j \in V$ 是节点 v_i 的父节点的可能性最大. 简言之, 条件熵 $H(x_i|x_j)$的值越小, 则节点 v_j 与节点 v_i 之间存在影响关系的可能性越大.

3.2 影响关系推断

条件熵衡量了某节点的感染状态结果是由指定父节点产生的概率, 我们可以使用它为传播网络中每个节点找到其可能性最大的若干个潜在的父节点. 此外, 我们还可以通过研究节点 v_i 的感染与其各潜在父节点的感染之间是否存在显著的相关性, 来判断哪些潜在父节点与节点 v_i 的影响关系明显较弱或几乎不存在.

具体来说, 对于节点 v_i 和节点 v_j 的感染状态变量 $v_i \in \{0,1\}$和 $v_j \in \{0,1\}$的任意取值, 如果等式:

$$p(X_i=x_i,X_j=x_j)=p(X_i=x_i)p(X_j=x_j) \tag{5}$$

成立, 则可认为节点 v_i 和节点 v_j 的感染状态取值相互独立. 互信息(mutual information)是一种常用的信息度量, 经常被应用于衡量变量之间的相关性. 然而, 传统的互信息衡量的是节点 v_i 和节点 v_j 的感染状态取值之间的相关性, 包括 0 和 1 两种状态, 而并非两个节点的感染事件(即只涉及感染状态为 1 的情况)之间的相关性. 因此, 我们对互信息的定义进行改进, 具体形式如下:

$$MI(X_i=1,X_j=1)=p(X_i=1,X_j=1)\cdot\log\frac{p(X_i=1,X_j=1)}{p(X_i=1)p(X_j=1)} \tag{6}$$

基于以上改进互信息, 我们可以估计潜在父节点与子节点的感染事件之间的相关性, 通过筛除感染行为与子节点相关性较弱的潜在父节点, 尽可能地避免加入强度太弱或根本不存在的节点间影响关系.

综上, 潜在的影响关系的推断分析过程可分为以下 4 个步骤.

(1) 初始化: 令 k=0, 并基于感染状态观测数据 S 计算传播网络中每两个节点 v_i 和 v_j 之间的条件熵;

(2) 查找潜在父节点: 令 k=k+1, 对于传播网络中每个节点 v_i, 找到使其条件熵数值最小的 k 个节点作为 v_i 的潜在的父节点;

(3) 计算相关性: 计算每个潜在的父节点与其子节点之间的改进互信息, 使用划分聚类算法, 即 K-means 算法(K=2), 将改进互信息按照数值的大小划分为两类, 并将改进互信息数值较小的一类中所对应的潜在父节点从各节点相应的潜在父节点集合中去除;

(4) 停止条件: 当传播网络中的所有节点都至少拥有一个潜在父节点时, 此推断过程停止; 否则, 返回步骤(2).

上述 4 个步骤持续地为传播网络中的节点添加可能性最大的潜在的父节点, 同时过滤掉相关性较弱的父节点. 重复以上操作, 直到每个节点都至少拥有一个潜在的父节点. 这是因为在真实的传播网络中, 尤其是稀疏网络, 一些节点的父节点数量很少, 甚至没有父节点, 所以使用以上停止条件能使推断出的影响关系结果的具有很高的召回率, 并在此基础上尽可能地防止添加目标传播网络中不存在的影响关系, 以保障较好的准确率.

3.3 推断结果优化

为了进一步提高推断潜在影响关系的准确率, 我们首先讨论哪种情况会造成原本与相关子节点无直接影响关系的节点成为潜在父节点, 然后介绍如何将这类伪潜在父节点筛除, 从而优化推断结果.

各节点与其推断出的潜在父节点表现出较强的感染相关性, 但这种相关性有时可能是由中介作用而非直接影响关系引起的. 例如, 即使一个节点 v_j 不能直接感染节点 v_i, 但是若节点 v_j 的某个或某些子节点恰恰是节点 v_i 的父节点, 那么节点 v_j 可以通过感染这些中间节点对节点 v_i 产生间接的影响. 这种间接的影响会造成推断结果中的节点 v_i 的潜在父节点集合有一定概率包含节点 v_j.

为了能够从所推断出的各节点潜在父节点集合中筛除这种伪潜在父节点, 我们可以考察节点及其潜在父

节点的感染事件之间是否具有直接相关性. 具体来说, 若要衡量节点 $v_i \in V$ 及其潜在父节点 v_j 的感染事件之间的直接相关性大小, 如节点 v_i 的其他潜在父节点至少有一个也处于感染状态时, 则需要排除这些处于感染状态的其他潜在父节点的中介作用. 这种情况可定义为 $X_{F_i \setminus v_j}=1$. 例如, 若节点 v_i 有 3 个父节点 v_j, v_k 和 v_ℓ, 则 $X_{F_i \setminus v_j}=1$ 代表以下 3 种情况: $(X_k=1, X_\ell=0)$, $(X_k=0, X_\ell=1)$ 和 $(X_k=1, X_\ell=1)$. 基于以上定义, 节点 v_i 和节点 v_j 的感染事件之间的直接相关性大小可以用改进的条件互信息(conditional mutual information)来衡量, 其具体计算公式如下:

$$CMI(X_i=1, X_j=1 \mid X_{F_i \setminus v_j}=1) = p(X_{F_i \setminus v_j}=1) \cdot p(X_i=1, X_j=1 \mid X_{F_i \setminus v_j}=1) \cdot \log \frac{p(X_i=1, X_j=1 \mid X_{F_i \setminus v_j}=1)}{p(X_{F_i \setminus v_j}=1) \cdot p(X_i=1, X_j=1 \mid X_{F_i \setminus v_j}=1)} \tag{7}$$

如果节点 v_j 是节点 v_i 的一个伪潜在父节点, 即节点 v_j 没有直接影响节点 v_i, 则相应的改进的条件互信息 $CMI(X_i=1, X_j=1 \mid X_{F_i \setminus v_j}=1)$ 的值将会很小, 通常趋近于 0. 因此, 对于推断得到的各节点的潜在父节点, 若其对应的改进 CMI 值趋近于 0, 则说明这个父节点为伪潜在父节点, 需将其从潜在父节点集合中筛除. 为了自动确定哪些改进 CMI 值趋近于 0, 我们在数值集合 $(c_1,\ldots,c_{m'},c_{m'+1})$ 上进行亲合度分析, 其中, m' 为潜在父节点的个数, $c_i(1 \leqslant i \leqslant m')$ 为第 i 个潜在父节点及其子节点之间对应的改进 CMI 值, 并令 $c_{m'+1}=0$.

在这个亲合度分析中, 数值 c_i 和 $c_j(i,j \in \{1,\ldots,m'+1\})$ 之间的亲和度定义如下:

$$A_{ij}=\begin{cases}\exp\left(-\dfrac{|c_i-c_j|}{t}\right), & i \neq j\\ 0, & i=j\end{cases} \tag{8}$$

其中, t 为常量, 为所有改进 CMI 数值 $(c_1,\ldots,c_{m'})$ 的均值. 亲和度分析旨在找到最接近于 0 的改进 CMI 值. 为此, 受到 Dominant Set 聚类框架[34]的启发, 我们提出了一个无参数的快速亲和度分析方法. 具体步骤如下.

(1) 初始化集合 $C=\{m'+1\}$ 以及向量 $V=(0,\ldots,0,1)^T$, 其中, 集合 C 用于记录最趋近于 0 的改进 CMI 值所对应的索引; 向量 V 为概率向量, 且其各元素总和为 1, 即 $\sum_{j=1}^{m'+1} v_j=1$, $v_j \geqslant 0$.

(2) 按照如下公式计算并找到在改进 CMI 值集合 $(c_1,\ldots,c_{m'},c_{m'+1})$ 中与集合 C 索引所的数值 $\{c_j \mid j \in C\}$ 亲和度最高的改进 CMI 值 c_{i^*} 所对应的索引 i^*:

$$i^*=\text{argmax}_i(V^TAe^i-V^TAV) \tag{9}$$

其中, e^i 为单位矩阵中的第 i 列, V^TAe^i 反映数值 c^i 和集合 $\{c_j \mid j \in C\}$ 中数值的平均亲和度, V^TAV 反映集合 $\{c_j \mid j \in C\}$ 内数值之间的平均亲和度.

(3) 如果 $i^* \in C$, 则停止亲和度分析, 并从推断得到的潜在父节点集合中筛除第 ℓ 个潜在父节点, 这里, $\ell \in C \setminus \{m'+1\}$; 否则, 更新集合 C, 使 $C=C \cup \{i^*\}$, 并按照如下公式(10)来更新概率向量 V 中的各元素后, 返回步骤(2):

$$V_i=\begin{cases}\dfrac{\omega(C,i)}{\sum_{j=1}^{|C|}\omega(C,l)}, & i \in C\\ 0, & i \notin C\end{cases} \tag{10}$$

其中, $\omega(C,i)$ 反映数值 c^i 和集合 $\{c_j \mid j \in C\}$ 中数值的平均亲和度与集合 $\{c_j \mid j \in C\}$ 内数值之间的平均亲和度的差值, 可以根据如下公式计算 $\omega(C,i)$ 的值:

$$\omega(C,i)=\begin{cases}1, & \text{if } |C|=1\\ \sum_{j \in C \setminus \{i\}} \Phi \cdot \omega(C \setminus \{i\}, j), & \text{otherwise}\end{cases} \tag{11}$$

$$\Phi = A_{ij} - \frac{1}{|C \setminus \{i\}|}\sum_{\ell \in C \setminus \{i\}} A_{j\ell} \tag{12}$$

以上亲和度分析的目的是, 找到一个包含数值 0 且内聚性极高的数值集合. 为实现这一目标, 该方法持续

向这个目标集合中添加能够使其内聚性增加的改进 CMI 值, 直到剩余的改进 CMI 值都不能增加目标集合的内聚性, 则停止向集合中添加数值. 因此, 此集合中的改进 CMI 值均趋近于 0, 且远远小于集合外的改进的 CMI 值. 而且, 该方法不需要为了判断最终结果的收敛而设定任何参数. 相反, 传统的 Dominant Set 聚类方法通常需要用户设定其方法的停止条件, 例如, 为向量 V 的变化量设定一个阈值[35], 而设定阈值的大小通常会影响这些方法的最终结果.

完成亲和度分析之后, 伪潜在父节点被从各节点的潜在父节点集合中移除. 余下的潜在父节点即为优化后的推断结果. 从每个潜在父节点出发, 做一条指向其对应子节点的有向边, 构成边集合 E, 即为 LDR 算法的输出.

3.4 复杂度分析

LDR 算法总共分两个主要阶段.

(1) 在影响关系推断阶段, 计算条件熵数的时间复杂度为 $O(\beta n^2)$, 其中, n 是目标传播网络中节点的数量, β是用于推断的历史传播过程的数量. 此外, 计算改进的互信息的时间复杂度为 $O(k\beta n)$, 执行 K-means 聚类算法的耗时为 $O(2k\tau n)$时间, 其中, k 为各节点潜在父节点数量的上限, τ是 K-means 聚类算法的迭代次数;

(2) 在推断结果优化阶段, 计算改进的条件互信息需要时间为 $O(k\beta n)$, 基于改进的条件互信息进行亲和度分析所需的时间复杂度为 $O(k^2n^2)$.

因此, 本文所提出的 LDR 算法的总体时间复杂度为 $O(\beta n^2+2k\beta n+2k\tau n+k^2n^2)$.

4 实验分析

4.1 实验设置

- 网络

我们采用 LFR 基准网络[36]作为人工合成的传播网络. 我们在生成 LFR 基准网络时, 节点个数设置为 n, 每个节点的平均度数设为 m/n, 其中, m 为网络中边的条数. 我们采用两组具有不同节点数量和平均度的 LFR 网络, 其参数设置见表 1. 此外, 我们还采用两个真实的传播网络, 即 NetSci 和 DUNF, 其中, NetSci[37]为表示论文作者之间合著关系的传播网络, 包含 379 个节点(代表科学家)及 1 602 条边(代表科学家之间的合著关系); DUNF[13]为表示微博用户之间互相关注及信息转发关系的传播网络, 包含 750 个节点(表示微博的用户)以及 2 794 条边(表示用户之间的关注和转发微博等).

表 1 LFR 网络

网络	节点数量	平均度
LRF1.1-1.5	100, 150, 200, 250, 300	4
LRF2.1-2.5	200	2, 3, 4, 5, 6

- 感染数据

为获得感染状态结果数据 S, 我们在人工和真实网络上模拟β次感染爆发过程, 观察并收集每次传播过程结束时, 网络中各个节点的感染状态数据. 在模拟感染传播的过程中, 我们随机选取初始感染节点, 其数量设置为∂. 另外, 在每次感染传播过程中, 我们设置感染传播概率为 0.3, 即每个已感染的父节点以 0.3 的成功概率对其尚未感染的子节点进行感染.

- 性能指标

为定量衡量本文算法所推断出的边集合 E 的准确性, 我们计算其推断出的边的 F-score(召回率和准确率的调和平均值). 该性能指标是衡量传播网络推断算法性能的常用指标[6−9,13,14,16,21,24], 计算公式如下:

$$F\text{-}score=\frac{2\cdot Precision\cdot Recall}{Precision+Recall},\ Precision=\frac{N_{TP}}{N_{TP}+N_{FP}},\ Recall=\frac{N_{TP}}{N_{TP}+N_{FN}},$$

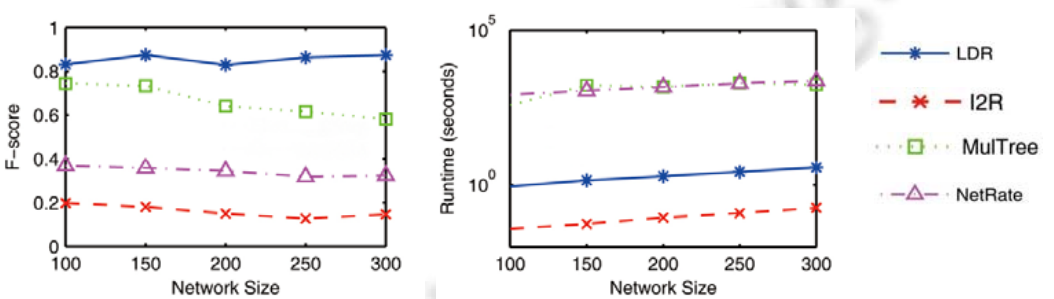
其中, N_{TP} 为真实存在且被推断出的边的数量, N_{FP} 为被推断出但并不真实存在的边的数量, N_{FN} 为真实存在但没有被推断出的边的数量.

- 对比算法

由于在现有基于时间信息的传播网络推断方法中, 基于嵌入空间的方法不能推断出显式的传播网络结构, 我们选取基于凸优化的方法中经典代表性算法 NetRate[12]以及基于子模性质的方法中的高性能代表性算法 MulTree[20]作为对比算法. 此外, 在现有不依赖时间信息的传播网络推断方法中, PATH 算法[11]需要所有沿传播路径连接的感染节点三元组在实际中太难以获得, 因此我们选取另一个不依赖时间信息的方法, 即 I2R[10]算法作为对比算法. 需要指出的是: 本文所提出的 LDR 算法和主要对比算法都推断出了目标传播网络的边,因此可以基于它们推断出的边来计算出各自确切的 *F-score* 用于性能评价. 但是对比算法中的 NetRate 算法未能推断出目标传播网络中哪些节点之间存在确切的边, 而是认为两两节点之间都可能存在边, 并推断每一条边的权重(即两两节点之间的传播概率). 如果按照两两节点都存在边来计算 NetRate 算法的准确性, 对该算法来说并不公平, 因此在性能比较时, 我们给 NetRate 算法一个特权: 由于权重较低意味着对应边上的两个节点存在影响传播关系的概率较低, 我们将权重超过某阈值的边视为 NetRate 算法推断出的边, 并计算这些边对应的 *F-score* 值; 同时, 通过不断变换所使用的阈值, 后验地找到 NetRate 算法能达到的最高 *F-score* 值作为该算法的准确性表现. 另一方面, 虽然本文 LDR 算法并不需要知道关于目标传播网络的先验知识, 但 MulTree 算法和 I2R 算法需要预先知道目标传播网络中边的数量, 因此在运行这两个对比算法时, 我们将传播网络中真实的边的数量 m 提供给它们.

4.2 传播网络大小的影响

为研究传播网络大小对算法性能的影响, 我们在 LFR1.1-1.5 上对算法进行了测试(β=150, ∂=0.15×n). 图 1(a)展示了算法在推断影响关系的准确性, 图 1(b)给出了它们的运行时间. 从图中可以看出: (1) LDR 算法的准确性明显优于其他测试算法; (2) 传播网络规模越大时, NetRate 和 MulTree 以及 I2R 的准确性越低; (3) I2R 执行速度最快(但其准确性亦最低), LDR 比 NetRate 和 MulTree 更高效; (4) 被测算法的运行时间一般随着传播网络规模的增大而升高.



(a) 准确性 (b) 时间效率

图 1 传播网络大小的影响

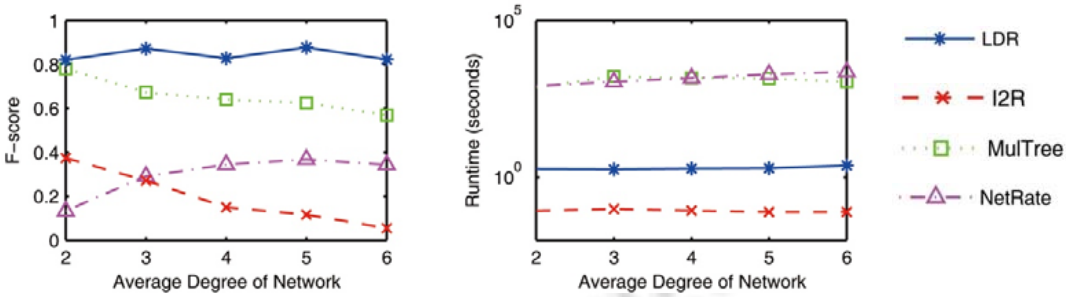
4.3 传播网络平均度的影响

为研究传播网络平均度对算法性能的影响, 我们在网络 LFR2.1-2.5 (β=150, ∂=0.15×n)测试算法. 图 2(a)和图 2(b)分别报告了各算法的准确性和运行时间, 从中我们可以得到以下观察结果: 传播网络平均度的变化度对测试算法的时间效率和 LDR 的准确性影响不大, 但对 NetRate 和 MulTree 以及 I2R 的准确性有显著影响.

4.4 传播过程数量的影响

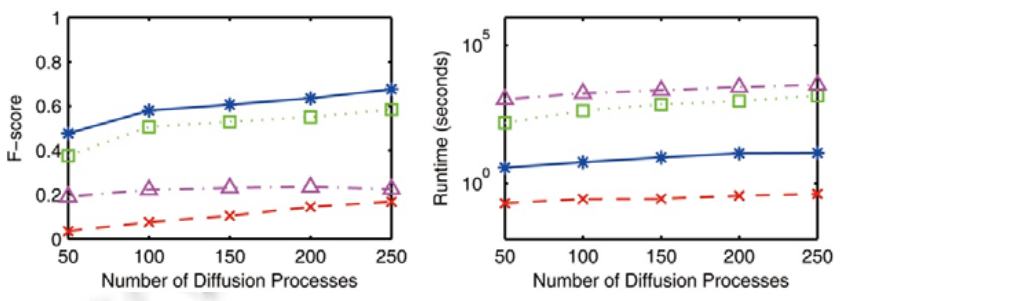
为研究传播过程的数量对算法性能的影响, 我们在真实网络上用不同传播过程数量β (β取值为 50 到 250 不等, ∂=0.15×n)测试算法性能. 图 3 报告了各算法的准确性和运行时间, 从中可以观察到: (1) 与对比算法相比, LDR 算法有更高的准确性和时间效率; (2) 传播过程越多, 所收集的感染结果数据越多, 有助于传播网络

推断算法获得更好的准确性.

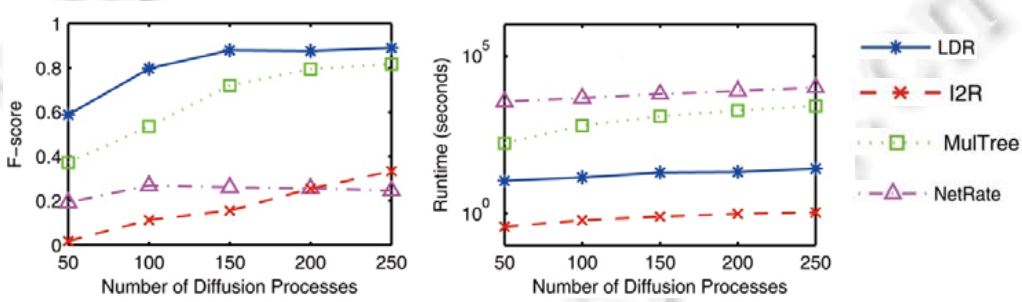


(a) 准确性 (b) 时间效率

图 2 传播网络平均度的影响



(a) 在 NetSci 上的准确性 (b) 在 NetSci 上的时间效率



(c) 在 DUNF 上的准确性 (d) 在 DUNF 上的时间效率

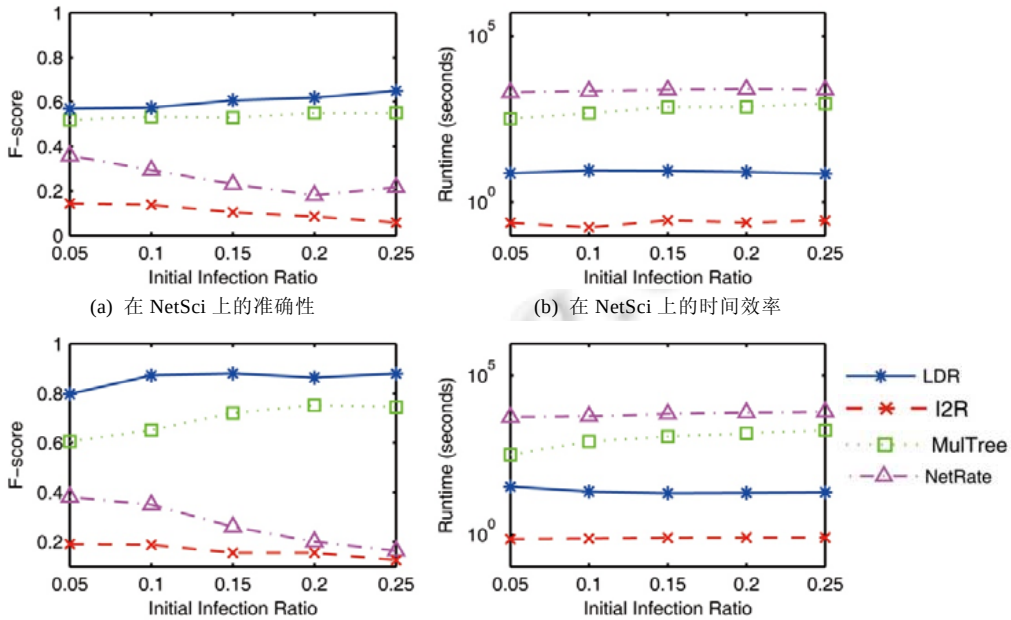
图 3 传播过程数的影响

4.5 初始感染比例的影响

为研究初始感染比率对算法性能的影响，我们在真实网络上用不同数量∂的初始感染节点测试算法(β=150, ∂取值为 0.05×n 到 0.25×n). 图 4 报告了各算法的准确性和运行时间，从中可以观察到：当初始感染节点越多时，往往能感染更多的节点，这时 LDR 和 MulTree 倾向于获得更加准确的推断结果，但 NetRate 和 I2R 的准确性则会有所降低.

5 结束语

本文研究了如何只利用传播结果来进行传播网络推断的问题. 为此，我们介绍了一个基于条件熵的概率生成模型来推断潜在影响关系，并讨论了如何利用节点感染的相关性来保障所推断影响关系的强度和真实性. 在人工网络和真实网络上的实验结果，证明了本文方法的有效性和运行效率.



(a) 在 NetSci 上的准确性 (b) 在 NetSci 上的时间效率

(c) 在 DUNF 上的准确性 (d) 在 DUNF 上的时间效率

图 4 初始感染率的影响

References:

[1] Gao S, Pang H, Gallinari P, *et al*. A novel embedding method for information diffusion prediction in social network bid data. IEEE Trans. on Industrial Informatics, 2017, 13(4): 2097−2105. [doi: 10.1109/TII.2017.2684160]

[2] Gomez-Rodriguez M, Leskovec J, Schölkopf B. Modeling information propagation with survival theory. In: Proc. of the ICML 2013. 2013. 666−674. http://proceedings.mlr.press/v28/gomez-rodriguez13.pdf

[3] Wallinga J, Teunis P. Different epidemic curves for severe acute respiratory syndrome reveal similar impacts of control measures. American Journal of Epidemiology, 2004, 160(6): 509−516. [doi: 10.1093/aje/kwh255]

[4] Leskovec J, Adamic LA , Huberman BA. The dynamics of viral marketing. ACM Trans. on the Web, 2007, 1(1): 5. [doi: 10.1145/1134707.1134732]

[5] Mehmood Y, Barbieri N, Bonchi F, *et al*. CSI: Community-level social influence analysis. In: Proc. of the ECML PKDD 2013. 2013. 48−63. [doi: 10.1007/978-3-642-40991-2_4]

[6] Han K, Tian Y, Zhang Y, *et al*. Statistical estimation of diffusion network topologies. In: Proc. of the ICDE 2020. 2020. 625−636. [doi: 10.1109/ICDE48307.2020.00060]

[7] Huang H, Yan Q, Chen L, *et al*. Statistical inference of diffusion networks. IEEE Trans. on Knowledge and Data Engineering, 2021, 33(2): 742−753. [doi: 10.1109/TKDE.2019.2930060]

[8] Huang H, Yan Q, Gan T, *et al*. Learning diffusions without timestamps. In: Proc. of the AAAI 2019. 2019. 582−589. https://aaai.org/ojs/index.php/AAAI/article/view/3833/3711

[9] Sun Y, Zhang Y, Yan Q, *et al*. Fast inference algorithm of diffusion networks without infection temporal information. Journal of Frontiers of Computer Science and Technology, 2019, 13(4): 541−553 (in Chinese with English abstract). [doi: 10.3778/j.issn.1673-9418.1807046]

[10] Amin K, Heidari H, Kearns M. Learning from contagion (without timestamps). In: Proc. of the ICML 2014. 2014. 1845−1853. http://proceedings.mlr.press/v32/amin14.pdf

[11] Gripon V, Rabbat M. Reconstructing a graph from path traces. In: Proc. of the ISIT 2013. 2013. 2488−2492. [doi: 10.1109/ISIT.2013.6620674]

[12] Gomez-Rodriguez M, Balduzzi D, Schölkopf B. Uncovering the temporal dynamics of diffusion networks. In: Proc. of the ICML 2011. 2011. 561−568. http://icml-2011.org/papers/354_icmlpaper.pdf

[13] Wang S, Hu X, Yu P, *et al*. MMRate: Inferring multi-aspect diffusion networks with multi-pattern cascades. In: Proc. of the KDD 2014. 2014. 1246−1255. [doi: 10.1145/2623330.2623728]

[14] Du N, Song L, Smola A, *et al*. Learning networks of heterogeneous influence. In: Proc. of NIPS the 2012. 2012. 2780−2788. http://papers.nips.cc/paper/4582-learning-networks-of-heterogeneous-influence.pdf

[15] Gomez-Rodriguez M, Leskovec J, Schölkopf B. Structure and dynamics of information pathways in online media. In: Proc. of the WSDM 2013. 2013. 23−32. [doi: 10.1145/2433396.2433402]

[16] Daneshmand H, Gomez-Rodriguez M, Song L, *et al*. Estimating diffusion network structures: Recovery conditions, sample complexity & soft-thresholding algorithm. In: Proc. of the ICML 2014. 2014. 793−801. http://proceedings.mlr.press/v32/daneshmand14.pdf

[17] Pouget-Abadie J, Horel T. Inferring graphs from cascades: A sparse recovery framework. In: Proc. of the ICML 2015. 2015. 977−986. [doi: 10.1145/2740908.2744107]

[18] Netrapalli P, Sanghavi S. Learning the graph of epidemic cascades. In: Proc. of the SIGMETRICS 2012. 2012. 211−222. [doi: 10.1145/2318857.2254783]

[19] Gomez-Rodriguez M, Leskovec J, Krause A. Inferring networks of diffusion and influence. In: Proc. of the KDD 2010. 2010. 1019−1028. [doi: 10.1145/2086737.2086741]

[20] Gomez-Rodriguez M, Schölkopf B. Submodular inference of diffusion networks from multiple trees. In: Proc. of the ICML 2012. 2012. 489−496. https://icml.cc/2012/papers/281.pdf

[21] Kurashima T, Iwata T, Takaya N, *et al*. Probabilistic latent network visualization: Inferring and embedding diffusion networks. In: Proc. of the KDD 2014. 2014. 1236−1245. [doi: 10.1145/2623330.2623646]

[22] Bourigault S, Lamprier S, Gallinari P. Representation learning for information diffusion through social networks: An embedded cascade model. In: Proc. of the WSDM 2016. 2016. 573−582. [doi: 10.1145/2835776.2835817]

[23] Bourigault S, Lagnier C, Lamprier S, *et al*. Learning social network embeddings for predicting information diffusion. In: Proc. of the WSDM 2014. 2014. 393−402. [doi: 10.1145/2556195.2556216]

[24] Abrahao B, Chierichetti F, Kleinberg R, *et al*. Trace complexity of network inference. In: Proc. of the KDD 2013. 2013. 491−499. [doi: 10.1145/2487575.2487664]

[25] Lokhov A. Reconstructing parameters of spreading models from partial observations. In: Proc. of the NIPS 2016. 2016. 3467−3475

[26] Sefer E, Kingsford C. Convex risk minimization to infer networks from probabilistic diffusion data at multiple scales. In: Proc. of the ICDE 2015. 2015. 663−674. [doi: 10.1109/ICDE.2015.7113323]

[27] Gan T, Han K, Huang H, *et al*. Diffusion network inference from partial observations. In: Proc. of AAAI 2021. 2021. 7493−7500. https://ojs.aaai.org/index.php/AAAI/article/view/16918/16725

[28] Shaghaghian S, Coates M. Bayesian inference of diffusion networks with unknown infection times. In: Proc. of the SSP 2016. 2016. [doi: 10.1109/SSP.2016.7551716]

[29] Shaghaghian S, Coates M. Online bayesian inference of diffusion networks. IEEE Trans. on Signal and Information Processing over Networks, 2017, 500−512. [doi: 10.1109/TSIPN.2017.2731160]

[30] Yan Q, Huang H, Gao Y, *et al*. Group-Level influence maximization with budget constraint. In: Proc. of the DASFAA 2017. LNCS 10177. 2017. 625−641. [doi: 10.1007/978-3-319-55753-3_39]

[31] Sun Y. Research of diffusion network inference technology based on infection state results [MS. Thesis]. Wuhan: Wuhan University, 2019 (in Chinese with English abstract).

[32] De Campos LM. A scoring function for learning Bayesian networks based on mutual information and conditional independence tests. Journal of Machine Learning Research, 2006, 7(7): 2149−2187. [doi: 10.1007/s10846-006-9082-0]

[33] Kempe D, Kleinberg J, Tardos É . Maximizing the spread of influence through a social network. In: Proc. of the KDD 2003. 2003. 137−146. [doi: 10.1145/956750.956769]

[34] Pavan M, Pelillo M. Dominant sets and pairwise clustering. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2007, 29(1): 167−172. [doi: 10.1109/TPAMI.2007.10]

[35] Bulò SR, Pelillo M, Bomze IM. Graph-based quadratic optimization: A fast evolutionary approach. Computer Vision and Image Understanding, 2011, 115(7): 984–995. [doi: 10.1016/j.cviu.2010.12.004]

[36] Lancichinetti A, Fortunato S, Radicchi F. Benchmark graphs for testing community detection algorithms. Physical Review E, 2008, 78(4). [doi: 10.1103/PhysRevE.78.046110]

[37] Newman MEJ. Finding community structure in networks using the eigenvectors of matrices. Physical Review E, 2006, 74(3): 036104. [doi: 10.1103/PhysRevE.74.036104]

附中文参考文献:

[9] 孙月明, 张运加, 颜钱, 等. 无需感染时间信息的传播网络快速推断算法. 计算机科学与探索, 2019, 13(4): 541–553. [doi: 10.3778/j.issn.1673-9418.1807046]

[31] 孙月明. 基于感染状态结果的传播网络推断技术研究 [硕士学位论文]. 武汉: 武汉大学, 2019.

赛影辉(1983－), 女, 博士生, 主要研究领域为数据挖掘.

侯叶俏(1996－), 女, 博士生, 主要研究领域为数据挖掘.

王明鑫(1995－), 女, 博士生, 主要研究领域为数据挖掘.

李翔翔(1982－), 女, 高级工程师, 主要研究领域为海量遥感数据处理与服务, 数据挖掘.

陈畅(1983－), 男, 博士, 讲师, 主要研究领域为复杂网络, 众包.

孙月明(1992－), 女, 硕士, 主要研究领域为数据挖掘

雷伯涵(1997－), 男, 博士生, CCF 学生会员, 主要研究领域为数据挖掘, 自然语言处理.

陈旭(1982－), 男, 博士, 副教授, 主要研究领域为海量遥感数据处理与服务, 数据挖掘.