

# 基于 TOPSIS 权重的社会网络影响力最大化

郭茂林, 孔 兵

(云南大学 信息学院, 云南 昆明 650091)

**摘 要:** 识别有影响力的用户和邮件是社会网络分析中最重要的主题之一, 如何寻找具有最小重叠和最大网络覆盖范围的种子节点集是社会网络研究中的一个重点问题。针对该问题已经提出了很多算法, 如贪心算法、CELF 算法、K-shell 算法和各种中心性度量排序算法等。种子节点之间的距离越近, 则节点之间的共同邻居节点越多, 造成覆盖范围的重叠。目前的算法往往不考虑种子节点间的距离和其覆盖范围的重叠, 导致最终的种子节点集质量不高。该文提出了一种新方法, 通过计算节点间的距离和重叠范围的综合权重, 以加权的“优劣解距离”(TOPSIS) 方法来选择有影响力的用户节点。与传统方法相比, 所选择的种子节点集合有更大的影响力散布。

**关键词:** 社会网络; 信息扩散; 影响力最大化; 客观权重; TOPSIS 方法

中图分类号: TP391

文献标识码: A

文章编号: 1673-629X(2022)07-0015-07

doi: 10.3969/j.issn.1673-629X.2022.07.003

## Maximization of Social Network Influence Based on TOPSIS Weight

GUO Mao-lin, KONG Bing

(School of Information Science and Engineering, Yunnan University, Kunming 650091, China)

**Abstract:** Identifying influential users and emails is one of the most important topics in social network analysis. How to find the seed node set with minimum overlap and maximum network coverage is a key problem in social network research. Many algorithms have been proposed to solve this problem, such as greedy algorithm, CELF algorithm, K-shell algorithm and various centrality metric sorting algorithm. The closer the distance between seed nodes, the more common neighbor nodes between nodes, resulting in overlapping coverage. Current algorithms often ignore the distance between seed nodes and the overlap of their coverage, resulting in the final seed node set of low quality. Therefore, we propose a new method that combines the objective weighting method and the TOPSIS method to select influential user nodes. This method aims to propose a solution by considering the above two goals. Compared with traditional methods, the set of seed nodes selected by the proposed method has greater influence spread.

**Key words:** social networks; information diffusion; influence maximization; objective weight; TOPSIS method

## 0 引 言

随着网络与社会生产活动的关系日益紧密, 社会网络的种类、数量和规模也呈现出了多样化、海量化和复杂化, 共同邻居共同邮件地址等重叠数据也越来越多。在这种背景下, 如何快速找到网络中的关键节点, 使网络信息以最大覆盖范围和最小重叠范围传播成了当下的重要研究问题之一。

影响力最大化问题可以定义为<sup>[1]</sup>: 在给定的网络中找到有限个数的活跃节点集, 经过特定的网络传播模型, 使得最终活跃节点的数量达到最大。求解影响力最大化问题的算法一般包括两个步骤, 一是评估给定网络上节点的影响力分布, 并对它进行排序; 二是在

排序基础上, 选择节点的最佳子集, 使网络传播具有最大的覆盖范围。

目前提出的影响力最大化算法主要是各种中心性度量排序的算法, 虽然该类算法时间快, 但是精度相对较低。如度中心性算法(DC)、度折扣算法(DD)、k-shell 算法<sup>[2]</sup>等, 选择具有最大散布的前  $k$  个节点为种子节点集, 不考虑种子节点之间可能存在的覆盖重叠。基于边际影响力进行排序的 IMRank 算法<sup>[3]</sup>, 通过对每个节点计算它的边际影响力来进行排序, 最终根据排序顺序来选择影响力最大的节点。上述算法在找影响力最大的节点集时, 没有考虑种子集成员之间的覆盖范围。目前的社会网络中, 由于人际交往的多样性

收稿日期: 2021-07-23

修回日期: 2021-11-25

基金项目: 国家自然科学基金项目(61762090, 62062066, 31760152); 云南省教育厅科学研究基金项目(2019J0005, 2021Y026)

作者简介: 郭茂林(1996-), 男, 硕士研究生, 从事社会网络分析、机器学习的研究; 通信作者: 孔 兵(1968-), 男, 副教授, 博士, 从事社会网络分析和机器学习的研究。

和广泛性,导致用户与用户之间的共同好友越来越多,所以在对相邻用户进行影响力建模时,可能会出现好友高度重叠,因此不一定会导致影响力最大化<sup>[4]</sup>。基于以上考虑,Zareie 等人提出了 MCIM 算法<sup>[1]</sup>,在考虑节点的直接影响和间接影响的基础上还考虑了节点的覆盖情况,但是,没有考虑节点的多个属性可能对种子节点选择有不同的影响,所有属性同等重要,与实际情况不相符,最终不一定能获得导致影响力最大化的节点集。

考虑到各种中心性度量排序算法的影响范围重叠问题和 MCIM 算法中属性权重不明确问题,该文提出了带权重多准则影响力最大化算法模型 (Multi-criteria Weight Influence Maximization, MWIM)。在该模型中,首先将影响力最大化问题建模为一个加权多准则问题,然后使用多种客观权重求解方法来计算各准则的权重。随后使用类似于理想解的优劣解距离法 (TOPSIS) 来指定开始扩散过程的最佳节点集。

## 1 理论基础

在社会网络影响力最大化研究中,一般将社会网络建模成图  $G(V, E)$ , 其中  $V$  表示网络中节点 (个体) 的集合,  $E$  表示网络中边 (关系) 的集合。网络中的节点有两种状态: 激活状态和未激活状态。激活状态表示个体接受某种行为或者事务, 未激活状态则表示个体未接受某种行为和事务。处于激活状态的节点对处于未激活状态的邻居节点存在影响, 它可以尝试去激活它的所有未被激活的邻居节点。被激活的节点又会影响其他处于未激活状态的节点。

TOPSIS 算法是一种常用的综合评价方法之一<sup>[5-6]</sup>, 它能充分利用原始数据的信息, 其结果能精确地反映各评价方案之间的差距。它用于从几个可能的选项中选择最佳的选项。由于这种算法的简单性和有效性, 吸引了许多研究者来解决经典的问题<sup>[5]</sup>。在决策问题中, 一些属性表示利润, 一些属性表示成本。正理想解和负理想解是在此算法中用于找到最佳替代方案的两个主要概念。在 TOPSIS 算法中, 利润属于极大型指标, 成本属于极小型指标, 积极的理想最大化利润标准, 最小化成本, 而消极的理想最大化成本标准, 最小化利润。TOPSIS 算法寻求选择与正理想解离最近与负理想解离最远的替代方案<sup>[6]</sup>。

## 2 基于 TOPSIS 的 MWIM 算法

为了消除种子集成员相同好友带来的精确度问题, 该文提出了基于 TOPSIS 的 MWIM 算法。它由三部分组成, 首先, 计算网络中节点的直接覆盖和间接覆盖属性值, 然后通过权重计算方法计算节点的直接覆

盖和间接覆盖权重, 最后再使用优劣解距离法 (TOPSIS) 找到网络的理想节点并将其作为新成员添加到种子集中。

在社会网络中, 对于具有更多连接关系的节点, 度中心性度量方式认为它们具有更高的影响力, 但是邻居节点的连接关系没有被考虑, 所以为了消除只考虑节点的度中心性的偏差, 所提出的方法还考虑了节点的二阶邻居数。记节点  $v$  的直接邻居数为  $DN_v = d_v$ , 其中  $d_v$  表示节点  $v$  的度数, 由于节点的二阶邻居重叠过多, 为了提高准确度, 使用信息熵来度量节点  $v$  的间接影响范围, 记  $IDN_v = E_1(v) + \alpha E_2(v)$ , 其中  $E_1(v)$  和  $E_2(v)$  表示节点  $v$  的一阶邻居和二阶邻居的度的熵值。

$$E_1(v) = - \sum_{u \in N_v} \frac{d_u}{d_v^1} * \log\left(\frac{d_u}{d_v^1}\right) \quad (1)$$

$$E_2(v) = - \sum_{u \in N_v} \frac{d_u^1}{d_v^2} * \log\left(\frac{d_u^1}{d_v^2}\right) \quad (2)$$

其中,  $d_v^1$  是节点  $v$  的一阶邻居的度数之和,  $d_v^2$  是节点  $v$  的二阶邻居的度数之和。  $a_v$  确定为  $a_v = \frac{d_v^2}{\max_{v_i \in V(d_i)} d_i^2}$ 。

在种子集的选择过程中, 不可避免地要考虑种子集间激活节点的重叠部分, 因为最终比较的是种子集激活的节点的总数, 即便种子集中的每个种子节点激活的节点数非常多, 但是由于重叠过多, 所以导致最终的节点激活总数增量非常小, 从而种子集节点在选择时, 需要考虑它们之间的距离。所以, 定义直接重叠 (DO) 和间接重叠 (IDO)。直接重叠是判断当前节点的邻居节点中, 有多少是种子集成员。而间接重叠是种子集与当前选择节点的直接邻居的重叠量。种子集中节点  $v$  的邻居数决定了节点  $v$  与该集合直接重叠的程度, 使用式 (3) 计算:

$$DO = \sum_{u \in N_v} S_u \quad (3)$$

其中,  $u$  是节点  $v$  的邻居节点, 如果  $u$  也是种子集成员, 则  $S_u$  的值为 1, 否则为 0。此外, 间接重叠表示节点  $v$  的直接邻居与种子集的邻居节点的重叠程度, 使用等式 (4) 计算:

$$IDO_v = \sum_{u \in S} S_u * |N_u \cap N_v| \quad (4)$$

其中,  $|N_u \cap N_v|$  表示节点  $u$  与  $v$  的邻居节点交集数, 等式 (5) 检查了所有已选种子集成员与节点  $v$  之间的重叠邻居数。在每次的迭代中使用等式 (3) 和 (4) 来动态更新节点  $v$  的直接覆盖和间接覆盖值, 其中较低的值表示最佳选择节点。具体步骤如下:

首先为所有节点计算四个特征的值, 然后按照等式 (5) 计算矩阵  $A$ :

$$A = \begin{bmatrix} DN_1 & IDN_1 & -DO_1 & -IDO_1 \\ \vdots & \vdots & \vdots & \vdots \\ DN_n & IDN_n & -DO_n & -IDO_n \end{bmatrix} \quad (5)$$

其中,矩阵  $A$  的第  $i$  行表示第  $i$  个节点的四个特征值。

但是 MCIM 算法没有考虑四个属性的权重,其中 DO 和 IDO 属性最大值与最小值差值很大,大多数节点的 DO 和 IDO 值为 0,这就表明了四个属性的变异程度是不一样的,所提供的信息量也不一样。为了更精确地基于四个属性来选择种子集,引入权重来考量四个属性对节点选择所做出的贡献大小是很有必要的,也在此基础上提出了 MWIM 算法。其中,在 MWIM 算法中,按照等式(6)来计算矩阵  $A$  :

$$A = \begin{bmatrix} DN_1 & IDN_1 & DO_1 & IDO_1 \\ \vdots & \vdots & \vdots & \vdots \\ DN_n & IDN_n & DO_n & IDO_n \end{bmatrix} \quad (6)$$

但是在计算权重时,为了避免因为人为的决策或者意图导致计算结果受主观偏好的影响,该文通过客观赋权法来计算权重。

在 MWIM 中,由于要考虑多个指标的综合评价,所以考虑了三种目前应用最广泛的客观赋权法、熵权法<sup>[7]</sup>、CRITIC 法<sup>[8]</sup>和标准离差法<sup>[9]</sup>。

如算法 1 所示,在 MWIM 算法中,首先初始化矩阵  $A$ ,其中  $DN_v$  表示节点  $v$  的直接邻居数,  $IDN_v$  表示节点  $v$  的间接邻居数,  $DO_v$  表示节点  $v$  与种子集节点之间的直接重叠数,  $IDO_v$  表示节点  $v$  与种子集节点之间的间接重叠数。在第 5 行中结合了中心性度量中的度中心性算法,通过求解节点的最大度数来确定种子集中的第一个节点  $u$ 。第 8 行通过删除种子节点  $u$  来更新矩阵  $A$ 。9~14 行通过遍历种子集节点来更新 DO 与 IDO 值。15 行说明了在 MWIM 算法中使用等式(7)来对 DO 和 IDO 值进行正向化。16 行中通过归一化操作对矩阵  $A$  中的数据消除量纲。第 17 行中的权重计算方法为上文介绍的熵权法,CRITIC 法和标准离差法。在第 18 行中,通过 TOPSIS 来计算剩余节点与种子集节点之间的方案距离并排序,以此找到的最优方案也就是种子节点。

在下一节中,将通过在 5 个真实网络上的实验来评估 MWIM 算法。其中, C\_MWIM 是使用 CRITIC 加权法的 MWIM 方法, E\_MWIM 是使用熵权法的 MWIM 方法, S\_MWIM 是使用标准离差法的 MWIM 方法。

$$x = \frac{x - \min}{\max - \min} \quad (7)$$

算法 1: MWIM 伪代码。

```

1. 输入:  $G < V, E >$ ,  $k // G$  是一个无向无权图,  $k$  是种子集大小
2. 输出: 种子集 SS
3.  $SS = \varnothing$ 
4. 初始化矩阵  $A = [DN_v \quad IDN_v \quad DO_v \quad IDO_v]$ 
5.  $s$  最大度的节点
6.  $SS = SS \cup \{s\}$ 
7. for  $i = 1$  to  $k-1$  do
8.  $A = A - \{s\}$ 
9.   for each  $v \in N_s$  do
10.  $DO_v = DO_v - 1$ 
11.   for each  $w \in N_v$  do
12.  $IDO_w = IDO_w - |N_w \cap N_s|$ 
13.   end for
14.   end for
15. 使用等式(7)对 DO 和 IDO 正向化
16. 对矩阵  $A$  归一化消除量纲得矩阵  $Z$ 
17. 计算矩阵  $Z$  中各个指标的权重  $w = (w_1, w_2, \dots, w_m)^T$ 
18. 通过 TOPSIS( $A$ ) 找到最优方案  $s$ 
19.  $SS = SS \cup \{s\}$ 
20. end for
21. 返回 SS

```

### 3 实验和对建议方法的评估

#### 3.1 实验网络

为了评估 MWIM 算法的性能,与 MCIM 算法、CELF 算法、K-shell 算法、IMRank 算法等方法进行了比较。由于 CELF 方法的时间复杂度过高,所以只在小型生成网络中与它进行比较。聚类系数指的是一个点的邻接点之间相互连接的程度,例如生活社会网络中,你的朋友之间的相互认识程度,所以聚类系数值越大,节点之间的互连程度也越大,共同邻接点也越多。因此针对聚类系数的大小,选取了聚类系数值过大,适中和过小的四个真实网络来对以上方法进行比较。四个网络特征如表 1 所示,其中 Hamstersters (HAM) 网络<sup>[10]</sup>表示基于网站 hamsterster.com 的用户之间的友谊和家庭关系的网络。Company (CPY) 网络<sup>[10]</sup>是使用来自大型欧洲研究机构的电子邮件数据生成的。Sport (SPT) 网络<sup>[10]</sup>收集有关 Facebook 页面的数据(2017 年 11 月)。Gnutella (GLA) 网络<sup>[11]</sup>是从 2002 年

表 1 网络信息

网络	节点数	边数	平均度数	平均二阶邻居数	聚类系数
Hamstersters	2 426	16 631	13.71	582.88	0.537 5
Gnutella	8 114	26 045	6.41	107.80	0.009 5
Company	14 113	52 100	7.00	163.23	0.239 2
Sport	13 900	86 800	12.48	480.73	0.276 1

8 月开始的 Gnutella 对等文件共享网络的快照序列。2002 年 8 月总共收集了 9 个 Gnutella 网络快照。节点表示 Gnutella 网络拓扑中的主机,边缘表示 Gnutella 主机之间的连接。

### 3.2 评估指标

为了模拟现实世界中的传播过程,使用流行病模型<sup>[12-13]</sup>来进行节点扩散建模。SIR 扩展模型早已用于评估种子集的影响传播能力。

流行病模型:该模型被广泛用于对社会网络中的消息传播过程进行建模<sup>[14]</sup>。该文将使用流行病模型中的 SIR 模型<sup>[15-16]</sup>来模拟扩展过程。在该模型中,每个节点可以处于以下三种状态之一:S(易感染)、I(已感染)和 R(已恢复)。首先,属于初始种子集的节点处于状态 I,其余节点都处于状态 S。在每个时间戳中,处于状态 I 的每个节点  $v$  都试图感染其邻居节点。为了这个目的,它以概率  $\beta$  感染状态为 S 的每个邻居节点,将它们移动到状态 I。节点  $v$  本身然后以概率  $\alpha$  进入状态 R。当状态为 I 的节点存在时,该过程将重复进行。最后,R 节点的数量表示初始种子集的影响范围。为了获得更准确的实验,SIR 过程需要运行多次,并将 R 节点的平均数量视为影响力的大小。

首先将种子集节点设置为状态 I,其余节点全部为 S 状态,为了消除少数次数的特殊性,将循环 500 次 SIR 过程,最后统计 R 状态的节点数,取平均值,数值

越大,说明算法效果越好。

### 3.3 在小网络上进行实验

为了对比 MWIM 算法与 CELF 算法的扩散效果,将使用空手道俱乐部网络(Zachary karate club)<sup>[10]</sup>,如图 1 所示。

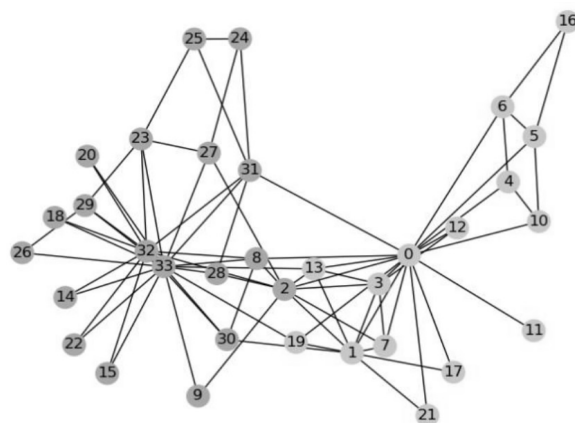
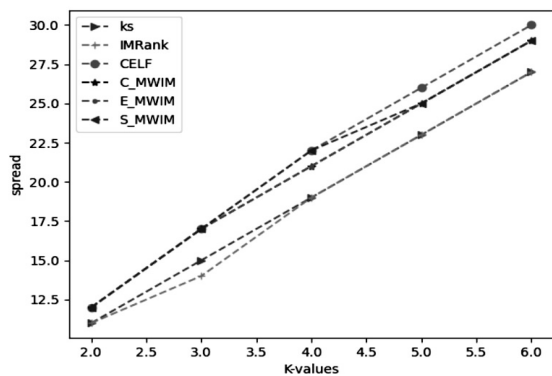


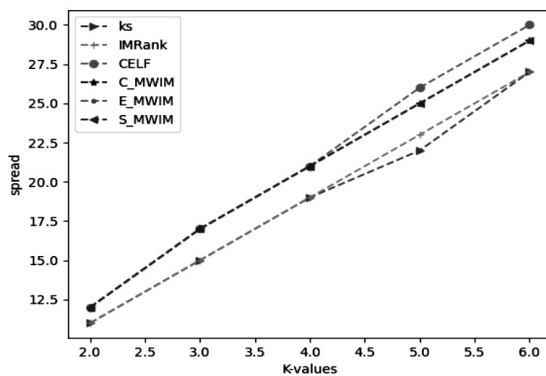
图 1 空手道俱乐部网络

该网络包含韦恩·扎卡里(Wayne Zachary)于 1977 年收集的大学空手道俱乐部成员之间的社会联系。该网络有 34 个节点 78 条边。该网络划分为两个社区,分别以节点 0 和节点 33 为中心。

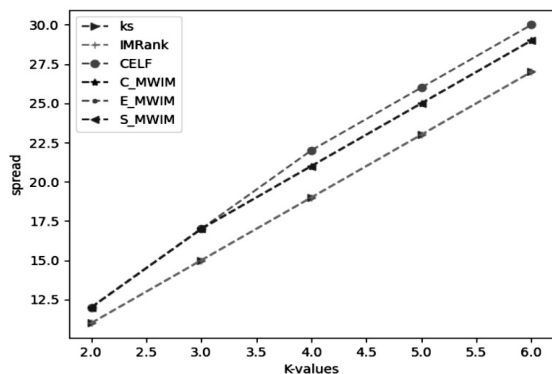
为了保证实验结果的可靠性,在使用 SIR 模型来评估种子集的传播情况时,分别迭代了 1 000,3 000,6 000 和 10 000 次,实验结果如图 2 所示。



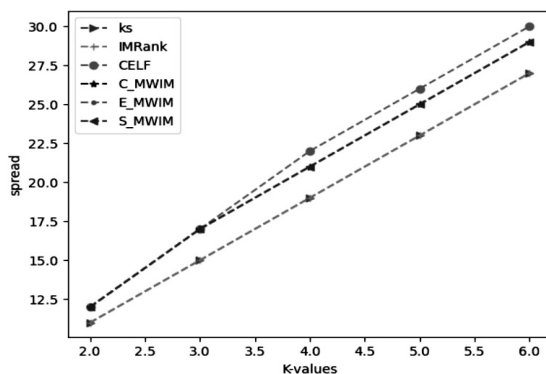
(a)1 000 次



(b)3 000 次



(c)6 000 次



(d)10 000 次

图 2 小网络扩散结果



实验结果表明,随着迭代次数的增加,各方法越来越稳定,MWIM 算法的三种权值方法所得结果完全一致,CELF 算法与其他方法的差值也保持不变,K-shell

算法和 IMRank 算法随着迭代次数的增加,也基本上完全重合。在表 2 中,给出了迭代次数为 10 000 时,各方法每次选择的种子集。

表 2 种子集大小  $k=2,3,4,5,6$  时各算法所选种子集

方法	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$
IMRank	0,23	0,23,14	0,23,14,20	0,23,14,20,22	0,23,14,20,22,15
K-shell	0,1	0,1,2	0,1,2,3	0,1,2,3,13	0,1,2,3,12,8
CELF	33,0	33,0,2	33,0,2,32	33,32,0,2,3	33,0,32,1,9,2
C_MWIM	33,0	33,0,32	33,0,32,31	33,0,32,31,30	33,0,32,31,30,29
E_MWIM	33,0	33,0,32	33,0,32,31	33,0,32,31,30	33,0,32,31,30,29
S_MWIM	33,0	33,0,32	33,0,32,31	33,0,32,31,30	33,0,32,31,30,29

3.4 在大网络上进行实验

针对不同聚类系数,比较在三种重叠程度下的 MWIM 算法与 K-shell、IMRank 和 MCIM 算法的扩散

效果。种子集的大小从 10 变到 50,每次增加 10。实验如图 3 所示。

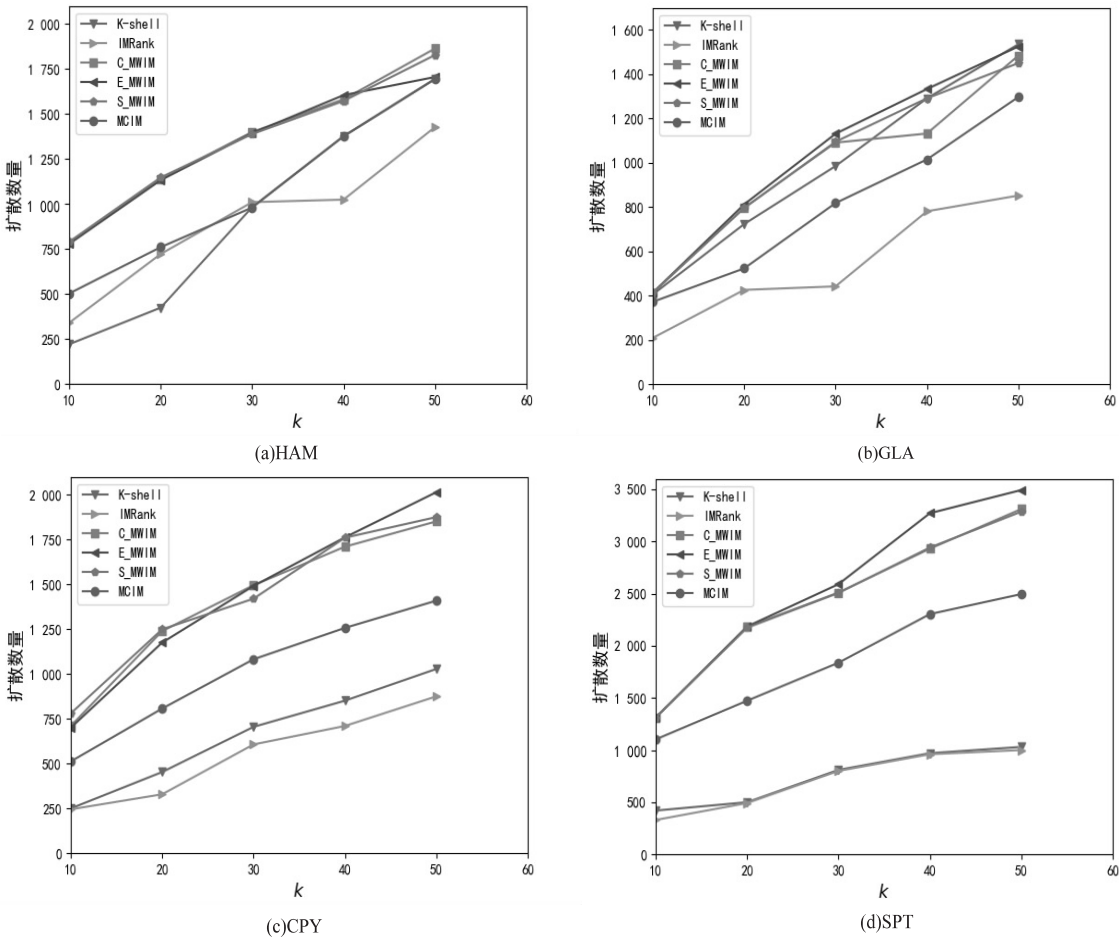


图 3 各算法扩散对比

图 3 结果表明,三种加权的 MWIM 方法所选择的种子集比 MICM 方法具有更高的影响散布。随着  $k$  值的增加,在 HAM、CPY、SPT 这三个网络上,MWIM 的散布情况更为突出,因为在这三个网络中,节点的邻接节点重叠区域较多,所以在等式(8)的矩阵  $A$  中直接重叠和间接重叠属性的非零值更多,在计算四个属性权重时,由于重叠属性的变异程度更大,所以对于重叠

属性给出了更高的权重。IMRank 方法的影响范围最低,因为它是去计算每个节点的边际影响力,再根据边际影响力排序,不停地迭代这个过程,最终的排列顺序就是节点的影响力大小排序,它忽略了节点的重叠。对于 K-shell 算法,它是通过不停地对节点分层,最后层数越高的节点认为它的影响力越大,通过对比 HAM 和 SPT、GLA 和 CPY 这两组实验结果可以发现,K-

shell 算法在 HAM 网络上随着  $k$  值的增加, 散布情况越来越靠近 MWIM 算法, 而在 SPT 网络上随着  $k$  值的增加, 与 MWIM 算法的散布差值越来越大。对比两个网络的属性发现, 主要的差异表现在聚类系数上, 聚类系数越大, 节点间结成团的程度越大, 比较之下, HAM 网络上的节点团程度更大, 所以 K-shell 算法在划分层数的时候, 在 HAM 网络上, 更能凸显 HAM 网络的层次。但是对比 GLA 和 CPY 网络, 随着  $k$  值的增加, GLA 网络的散布情况越来越与 MWIM 算法相近, 而在 CPY 网络上, 散布差值越来越大, 它们的主要差异也是表现在聚类系数上, 由于 GLA 网络的聚类系数过小, 节点与邻居节点的互联程度也非常小, 所以 K-shell 算法在对它进行层次划分时, 层数会很少, 并且每一层的节点数会很多, 导致节点间的重要程度都不大, MWIM 算法在对 GLA 网络计算权重时, 由于节点之间离散度大, 重叠程度小, 所以权重差异性也不大, 所以会出现图 3(b) 的情况。

通过实验结果可以看出, 使用熵权法和标准离差法的赋权结果较为接近, 因为熵权法和标准离差法的基本思路非常相似, 都是通过指标变异的大小来确定

权重。对于 CRITIC 赋权法, 它还额外考虑了各指标之间的冲突性, 所以在聚类系数更大的网络上, 它的结果更好一些。

### 3.5 时间对比实验

在这个实验中, 将比较 MWIM 算法与其他算法在不同  $k$  值时的平均运行时间。实验结果如图 4、图 5 所示。图 4 结果表明, CELF 算法平均运行时间较长, 不适用于大型网络的种子集求解。在小网络上 MWIM 算法与 K-shell 算法和 IMRank 算法的平均运行时间相差不大。

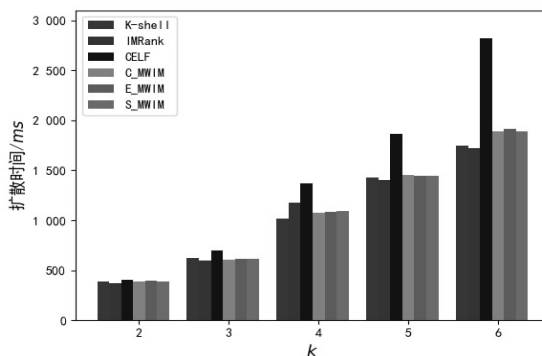


图 4 不同算法在空手道俱乐部的运行时间对比

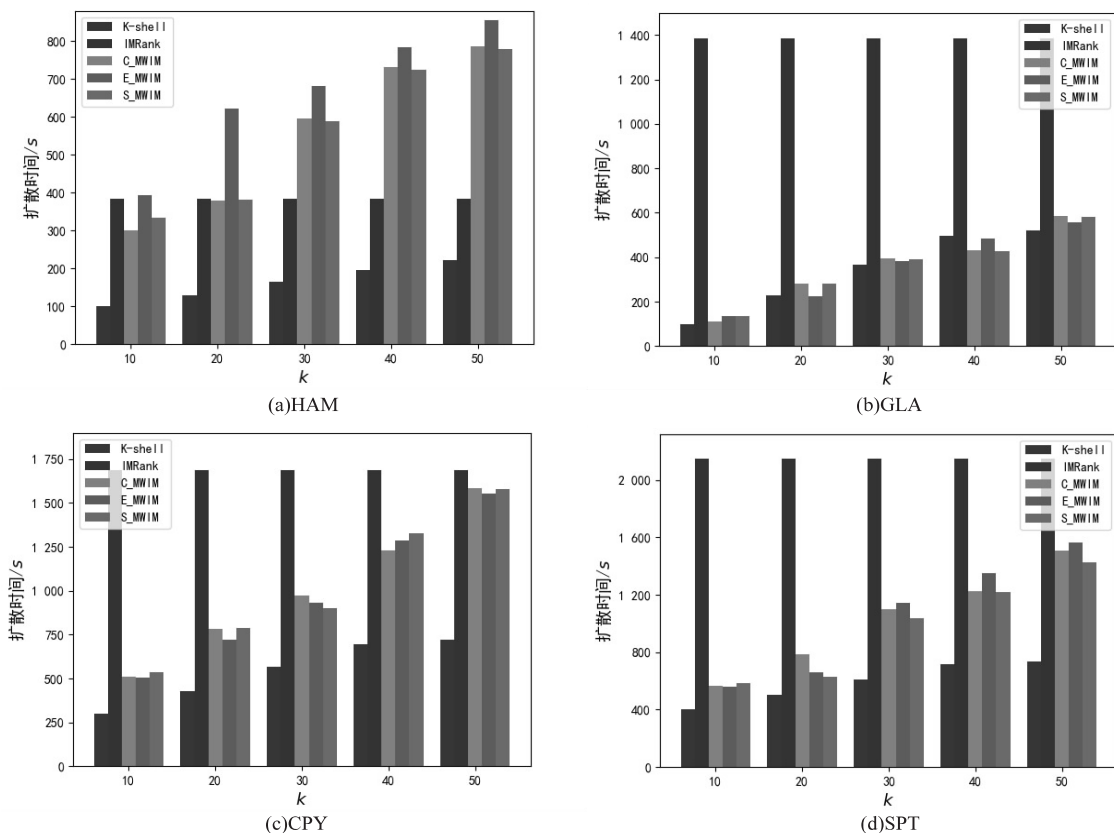


图 5 各算法平均运行时间

图 5 的结果表明, 在 HAM 网络上, MWIM 算法的平均运行时间大于 K-shell 算法和 IMRank 算法, 因为 HAM 网络的聚类系数过大, 所以节点与节点之间的联系更为紧密, 在计算重叠部分时所需时间更多。在

GLA, CPY 和 SPT 网络上, IMRank 算法的平均运行时间最大。其中, 在四个网络中 K-shell 算法的平均运行时间都是最小。由于 IMRank 算法是通过遍历一遍所有节点, 求每个节点的边际影响力, 再根据边际影响

力进行排序,由于计算所有节点的边际影响力的时间远远大于选择种子集的时间,所以 IMRank 算法随  $k$  值的变化不明显。在 GLA 网络上,  $K$ -shell 算法与 MWIM 算法的平均运行时间相差不大,因为 GLA 网络的聚类系数非常小,所以节点与节点之间的联系稀疏,导致节点间的重叠部分少, MWIM 算法在计算重叠部分时所花时间就大大减少。

#### 4 结束语

近年来,信息在社会网络上的传播引起了极大的关注,选择有影响力的用户集对于信息的传播尤为重要,如何选择用户即成了一个关键问题。针对不同的网络,学者们提出了不同的方法。在选择有影响力的用户集时应该不单单只看用户的邻居数量,还应该考虑邻居的影响范围以及他们影响的重叠量和距离。该文提出了一种 MWIM 方法来选择具有最大影响力的用户集,在该方法中,不仅考虑了直接邻居的影响范围,还考虑了间接邻居的影响范围,在考虑如何度量各个因素的影响力大小时,引入客观权重来解决这个问题。在度量了各个因素的权重之后,使用 TOPSIS 方法来选择一组当前最优的节点作为后续扩展的种子集节点。在各种真实网络上进行了一系列不同的实验,以研究和评估该方法的准确性和效率。使用 SIR 模型来对种子集进行模拟扩散,结果表明 MWIM 算法在选择种子集时,比其他算法更为精确。在使用小网络进行影响力种子集选择实验中, MWIM 算法的结果更接近于 CELF 算法,在每一轮的种子集选择中,都能准确地选取到重要节点。在提出的各种算法中,种子集大小  $k$  已被视为影响力最大化问题的输入值,尚未指定最佳  $k$  值的标准。但是,适当的  $k$  值可以使信息传播所需消耗和收益达到一个最佳值。在未来的工作中可以考虑最佳  $k$  值的选取研究,还有在聚类系数特别小的网络上怎么对 MWIM 算法进行优化。

#### 参考文献:

- [1] ZAREIE A, SHEIKHAHMADI A, KHAMFOROOSH K. Influence maximization in social networks based on TOPSIS [J]. Expert Systems with Applications, 2018, 108: 96–107.
- [2] KITSACK M, GALLOS L K, HAVLIN S, et al. Identification of influential spreaders in complex networks [J]. Nature Physics, 2010, 6: 888–893.
- [3] CHENG S, SHEN H W, HUANG J, et al. IMRank: influence maximization via finding self-consistent ranking [C]//37th international ACM SIGIR conference on research and development in information retrieval. QLD, Australia: ACM, 2014: 475–484.
- [4] BAO Z K, LIU J G, ZHANG H F. Identifying multiple influential spreaders by a heuristic clustering algorithm [J]. Physics Letters A, 2017, 381(11): 976–983.
- [5] HU J, DU Y, MO H, et al. A modified weighted TOPSIS to identify influential nodes in complex networks [J]. Physica A Statistical Mechanics and Its Applications, 2016, 444: 73–85.
- [6] ALI A, MAHESH T M, ERFAN A. Role of public participation in sustainability of historical city: usage of TOPSIS method [J]. Indian Journal of Science and Technology, 2012, 5(3): 2289–2294.
- [7] ZOU Z H, YI Y, SUN J N. Entropy method for determination of weight of evaluating indicators in fuzzy synthetic evaluation for water quality assessment [J]. Journal of Environmental Sciences, 2006, 18(5): 1020–1023.
- [8] DIAKOULAKI D, MAVROTAS G, PAPAYANNAKIS L. Determining objective weights in multiple criteria problems: the critic method [J]. Computers and Operations Research, 1995, 22(7): 763–770.
- [9] 王 昆, 宋海洲. 三种客观权重赋权法的比较分析 [J]. 技术经济与管理研究, 2003(6): 48–49.
- [10] RYAN A R, NESREEN K A. An interactive data repository with visual analytics [J]. ACM SIGKDD Explorations Newsletter, 2016, 17(2): 37–41.
- [11] LESKOVEC J, KLEINBERG J, FALOUTSOS C. Graph evolution: densification and shrinking diameters [J]. ACM Transactions on Knowledge Discovery from Data, 2007, 1(1): 2–es.
- [12] BUSCARINO A, FORTUNA L, FRASCA M, et al. Disease spreading in populations of moving agents [J]. EPL (Europhysics Letters), 2008, 82(3): 38002.
- [13] ZHOU J, CHUNG N N, CHEW L Y, et al. Epidemic spreading induced by diversity of agents' mobility [J]. Physical Review E, 2012, 86(2): 026115.
- [14] 罗荣桂, 江 涛. 基于 SIR 传染病模型的技术扩散模型的研究 [J]. 管理工程学报, 2006, 20(1): 32–35.
- [15] HUANG C Y, LEE C L, WEN T H, et al. A computer virus spreading model based on resource limitations and interaction costs [J]. Journal of Systems & Software, 2013, 86(3): 801–808.
- [16] PASTOR-SATORRAS R, VESPIGNANI A. Epidemic spreading in scale-free networks [J]. Physical Review Letters, 2000, 86(14): 3200–3203.