

基于合并影响概率的社交网络影响最大化算法

周 飞,高茂庭

(上海海事大学 信息工程学院,上海 201306)

摘 要: 针对大型社交网络影响最大化算法时间复杂度较高,并且节点影响覆盖率较低的问题,提出一种新的影响力最大化算法。采用 PageRank 算法选择影响力较高的节点作为备用种子,通过统计备用种子对潜在可激活节点的激活轮次和激活次数来计算其合并影响概率,并采用遗传算法从中选择合并影响概率最大的 k 个结果作为种子节点。仿真结果表明,与 DegreeDiscount、PageRank 等算法相比,该算法能获得较好的节点选取效果。

关键词: 社交网络;影响最大化;合并影响概率;遗传算法;独立级联模型

中文引用格式:周 飞,高茂庭. 基于合并影响概率的社交网络影响最大化算法[J]. 计算机工程,2018,44(5):188-193,200.

英文引用格式:ZHOU Fei,GAO Maoting. Influence Maximization Algorithm for Social Network Based on Combined Impact Probability[J]. Computer Engineering,2018,44(5):188-193,200.

Influence Maximization Algorithm for Social Network Based on Combined Impact Probability

ZHOU Fei,GAO Maoting

(College of Information Engineering, Shanghai Maritime University, Shanghai 201306, China)

[Abstract] In order to solve the problem of high time complexity and low impact coverage of nodes in influence maximization algorithm, a new influence maximization algorithm is proposed. The PageRank algorithm is used to select the nodes with higher influence as the standby seeds. Then, the combined impact probability of the standby nodes is calculated by counting the number of active rotations and activations of the nodes which can be activated, and k nodes with the largest probability of the combined impact are selected as the seed nodes by the Genetic Algorithm (GA). Simulation results show that compared with DegreeDiscount, PageRank and other algorithms, the algorithm can obtain better nodes selection effect.

[Key words] social network; influence maximization; combined impact probability; Genetic Algorithm (GA); Independent Cascade Model (ICM)

DOI:10.19678/j.issn.1000-3428.0046702

0 概述

随着互联网技术的迅猛发展,社交网络服务(Social Network Service, SNS)作为互联网应用发展的必备要素,不再局限于信息传递,而是与沟通交流、商务交易类应用融合,借助其他应用的用户基础,形成更强大的关系链,从而实现对信息的广泛、快速传播。鉴于社交网络影响力的传播特性,信息在社交网络传播中具有“口碑效应”,即当某用户接受一新鲜事物时,他通常会将该事物推荐给他的朋友,当他的朋友接受的时候就实现了信息的有效扩散。于是,在日常生产生活中决策者们便利用“口碑效应”在社交网

络中进行产品信息宣传,实现影响最大化在网络营销、舆情监控等方面的实际应用。社交网络的流行也为“病毒式营销”提供了天然的营销网络,因此,如何从众多网络节点中寻找若干较少节点,使得影响力能够最广泛地扩散成为一个研究热点。

已知社交网络由 M 个节点和 N 条边所组成的有向图 $G(V, E)$ 表示。社交网络影响最大化问题(Influence Maximization Problem, IMP)由文献[1]提出,IMP 问题就是如何从 M 个节点中选取 K 个种子节点,让这个 K 种子节点在初始时刻处于激活状态,通过网络传播模型尝试激活其他当前状态是未激活的节点,最终使得被激活成功的节点数最多的问题。

基金项目:国家自然科学基金(61202022)。

作者简介:周 飞(1993—),男,硕士研究生,主研方向为数据挖掘、数据分析;高茂庭,教授、博士。

收稿日期:2017-04-10 **修回日期:**2017-05-23 **E-mail:** phil_chow@outlook.com

针对该问题,本文提出一种基于合并影响概率的社交网络影响最大化算法。

1 相关研究

为了解决影响最大化问题,文献[2]将影响最大化问题归纳为离散最优问题,并提出了近似可达最优解 63% 的爬山贪心算法,运用多次蒙特卡洛模拟获得影响范围,取最优解,但对于大规模社交网络,这种贪心算法的时间复杂度太高。针对此问题,文献[3]提出了改进的贪心算法 CELF,利用网络传播的子模性,延迟计算边际收益,将时间效率提高了数百倍。在此基础上,文献[4]利用堆特性对 CELF 算法做出改进并提出了 CELF++ 算法。文献[5]提出 NewGreedy 算法,在独立级联传播模型下,以 $1-p$ 的概率去除原图中的边,再迭代考虑子图的最大影响力。MixGreedy 算法^[5]结合 NewGreedy 算法和 CELF 算法,仿真实验表明,其性能略好于 NewGreedy。文献[5]亦在 Degree 算法^[6]的基础上提出 DegreeDiscount 算法,性能也有所提升。网页排名的 PageRank 算法^[7]也被运用于寻找网络影响力节点中。PMIA 算法^[8]提供了稳定的传播范围,且运行速度比贪心算法提升了大约 3 个数量级,但由于在本地计算节点树结构,运行时需要耗费较大内存。文献[9]提出 CGA 算法,采用分治思想,拆分数据集,对各个子集并行计算。文献[10]在 CELE 的基础上使用上界逼近法减小了算法响应时间。UGGreedy 算法^[11]在去除不重要节点简化网络结构后再使用贪心算法求解,但算法时间复杂度依然相对较高。文献[12]提出 k-核概念,并考虑节点间影响区域重叠现象提出核覆盖 CCA 算法,认为影响重叠会使得影响力难以扩散,带来的边际受益很小。CCA 算法优先选择距离参数 d 内影响重叠较少的节点。然而,在独立级联的模型下,重叠部分的节点被影响次数要多于非重叠部分节点,因此,重叠部分节点相对被影响的概率更大,继而可以影响到其他后续节点。文献[13]结合网络在线传播和现实社会中口口相传的特性建立模型,但也正因如此算法扩展性较差。BCIM 算法,先使用 PageRank 选取备用节点,再使用动态规划的方法获取最优解,其不足之处是只考虑到近距离邻居的影响,虽然在算法时间上有较大提高,但是会出现部分影响力在传播过程中丢失的现象。

为了更好地解决影响最大化问题,且考虑到现有算法中存在贪心算法时间复杂度过高,节点间影响区域重叠,算法可扩展性以及传播过程中只考虑近距离传播而牺牲影响力间接传播来提高算法时间性能等问题。本文在第 4 节提出基于合并影响概率的社交网络影响最大化算法 (Influence

Maximization with Combined Impact Probability, CIPIM),在沿用 BCIM 算法中先使用 PageRank 选取备用种子节点,再在计算备选种子节点合并影响概率的基础上,使用遗传算法 (Genetic Algorithm, GA) 解决全局优化问题。

2 传播模型

寻找社交网络影响最大化节点往往需要借助于网络传播模型,通常情况下将社交网络表示为由 M 个节点和 N 条边所组成的有向图 $G(V, E)$,其中,节点表示社交网络中的个体,有向边表示个体之间的关系,如 Twitter 中用户之间的关注关系。线性阈值模型 (Linear Threshold Model) 和独立级联模型 (Independent Cascade Model, ICM) 是 2 种常用的网络传播模型。

2.1 线性阈值模型

在线性阈值模型^[14]中,任意节点 v 都有一个阈值 $\theta_v \in [0, 1]$,该阈值表示当前节点受其邻居节点影响的难易程度, θ_v 越大越难被影响。令 U 为节点 v 的邻居节点 u 的集合, I_{uv} 表示 u 对 v 的影响度,那么当 $\sum_{u \in U} I_{uv} \geq \theta_v$ 时,即表示节点 v 受到了影响,由不活跃状态转变为活跃状态。但是,在现实社交网络中很难确定每个节点的阈值,因此很多时候将其设置为服从 0-1 分布的随机数。

2.2 独立级联模型

在独立级联模型^[15]中,任意一条边 $\langle u, v \rangle \in E$ 都有一个 $p_{uv} \in [0, 1]$,表示节点 u 通过边 $\langle u, v \rangle$ 影响节点 v 的概率。该模型中,只有在当前节点被激活后才有一次机会去激活其邻居节点。假设 u 在时间 t 被激活,那么在单步时间内, u 可以尝试去激活它的邻居 v 。如果 v 被激活,那么 v 将在时间 $t+1$ 变成激活状态。时间 $t+1$ 之后, u 将不再尝试激活其邻居节点。当若干单步时间后,如果不存在激活可能性的时候,传播过程结束。在简单的独立级联模型中,通常将 p_{uv} 设置为常量,或是从 $\{0.1, 0.01, 0.001\}$ 中随机选取。独立级联模型更侧重于影响力的传播过程,在现实工作中应用更为广泛,因此本文选用独立级联模型作为传播模型。

3 影响最大化算法 CIPIM

影响最大化问题就是要从 M 个节点中选取影响传播影响范围最大的 K 个种子节点。但在实际社交网络中存在大量低影响力用户,在信息传播中几乎没有任何贡献,因此,他们不能作为种子节点。从减少种子选取范围上考虑,先使用 PageRank 算法计算 M 个节点的 PageRank 值,从中选取排名靠前的节

点作为备选种子集合,再对各备选种子节点进行合并影响概率预计算,最后使用遗传算法进行全局优化挑选出 K 个种子节点。

3.1 种子选取范围的减少

意见领袖通常是网络社区中的活跃分子,是信息的积极传播者,能够引起大量关注并影响社区中的舆论导向。在线社交网络通常采用 PageRank 值来表示用户的影响力大小,值越大则影响力越大。社交网络中还存在着大量的信息接收者,但单纯的接受者对信息传播的贡献却非常小。为了分析选取高影响力用户作为备用种子节点的占比规律,对 Wiki-Vote 数据集^[16] 7 115 个节点按照 PageRank 值进行排序,归一化处理各节点 PageRank 值并计算可影响范围占比情况,考虑图像显示效果和方便观察,截取前 1 000 个节点,如图 1 所示。1 000 位之后的图像延续图 1 后半段走势,平缓递增与递减。

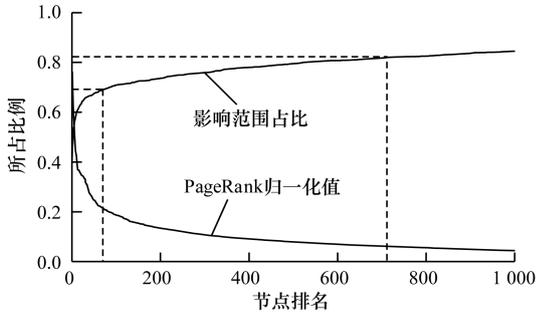


图 1 前 1 000 名 PageRank 值及影响范围占比情况

图 1 显示,选取 1/10 的节点就可达到超过 80% 的影响范围,仅需 1% 的节点即可达到 60% 以上的影响范围。因此,本文从减小种子节点选取范围出发,选取具有较高 PageRank 值的节点作为备选种子节点。考虑到当 K 值较小时(比如 $K = 10$),并不需要从前 500 个甚至很多的节点中挑选出 10 个种子节点,仅从前 100 个节点来看完全可以达到预想效果。同时,为了防止使用固定数量的备选种子可能会造成的局部最优情况,本文使用线性规则来选取备用种子节点,即选取 PageRank 值排名靠前的 hK 个节点作为备选种子节点,为了方便计算在实验中将 h 值设为 10。这样,不仅从数量上减少运算时间,而且在一定程度上消除可能存在的局部最优情况。

3.2 备选种子合并影响预计算

文献[17]在数据集 DBLP 上证实了六度分割理论猜想:每个人最多通过 6 个人就可以认识一个陌生人。同样,在社交网络信息传播的过程中,也仅需几次即可将信息传播开。图 2 是社交网络中种子节点 $A、B$ 影响传播的局部路径简单传播模型,其中, t 表示当前传播次数。

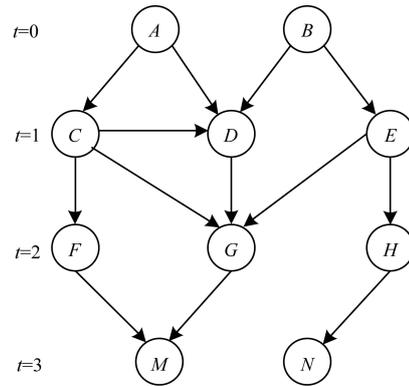


图 2 社交网络信息传播路径

在图 2 中,设每条有向边传播概率相同,为常量 p ,那么种子节点 A 通过边 $\langle A, C \rangle$ 激活节点 C 的概率即为 p ,节点 C 被激活后,就有 p 的概率通过边 $\langle C, F \rangle$ 影响到 F 节点,故节点 A 通过边 $\langle A, C \rangle$ 和 $\langle C, F \rangle$ 激活节点 F 的概率为 p^2 。节点 D 的影响来源于种子节点 $A、B$ 以及节点 C 的传递影响,那么它可被以 p 概率激活 2 次,以 p^2 概率激活一次,于是,节点 D 被激活的概率为 $1 - (1 - p)^2 + p^2$ 。

备选种子合并影响预计算,是对备选种子集中每个备选种子进行一次节点自身传播范围内可被激活节点次数及轮次统计。对于种子节点 A ,其可激活节点集为 $\{C, D, F, G, M\}$,且 $t = 1$ 的有 $\{C, D\}$, $t = 2$ 的有 $\{D, F, G\}$, $t = 3$ 的有 $\{G, M\}$,其中,当 $t = 3$ 时有节点 G 的原因是存在一条 $A-C-D-G$ 通路。对于种子集合 $\{A, B\}$ 来说,由于它们之间有共同影响部分,故它们的影响概率并不是节点 A 和 B 的概率简单相加。因此,在最终计算种子集合的影响概率时,先要分别统计各种子节点的预处理结果,再合并计算它们的影响概率。

在不同传播概率 p 下,若以不同传播次数 t 分别尝试 500 次激活,通过公式 $P = 1 - (1 - p^t)^{500}$ 可计算出节点被激活概率。例如,当 $t = 3$ 时,节点被激活的概率为 p^3 ,假设该节点被激活 500 次,则该节点在 $t = 3$ 下被激活概率为 $1 - (1 - p^3)^{500}$,如表 1 所示。

表 1 500 次激活下节点被激活概率

t	$p = 0.06$	$p = 0.03$	$p = 0.01$
2	0.835 237	0.362 501	0.048 773
3	0.102 383	0.013 409	0.000 500
4	0.006 459	0.000 405	0.000 005

如表 1 所示,当 $t = 3, p = 0.01$ 时,仍有万分之五的概率能够激活节点,而当 $t = 3, p = 0.06$ 时被激活概率达到约 0.1。当 $t = 4, p = 0.01$ 时,节点几乎不可能被激活,而 $p = 0.06$ 时也仅仅只有 0.006 的概率。因此,本文针对文献[13]中只考虑近距离($t = 2$)传播的问题,将种子节点可影响步数调整为 $t = 3$ 步,

虽然在一定程度上加大了算法的时间复杂度,但是其传播概率计算更为准确合理。

设 $G(V, E)$ 为有向图, $seed$ 表示某种子节点, t 表示当前传播轮次, 该备选种子预处理算法 (Alternative Seed Preprocess Algorithm, ASPA) 采用图深度优先遍历策略, 算法描述如下:

输入 $t, seed, G(V, E)$

输出 各种子节点统计信息

执行步骤:

1) 如果 $t > 3$, 返回, 否则转到 2)。

2) $F_{seed} \leftarrow$ 节点 $seed$ 所有未尝试激活的邻居节点, 转到 3)。

3) $SR(seed, times) \leftarrow F_{seed}$, 转到 4)。

4) 标记节点 $seed$ 已尝试激活, 转到 5)。

5) 对于 $\forall u \in F_{seed}$, 递归计算 $ASPA(t + 1, u, G)$ 。

6) 返回 SR 。

其中, 步骤 1)、步骤 3)、步骤 4) 的时间复杂度都是 $O(1)$, 步骤 2) 的时间复杂度为 $O(n)$, G 图以类似邻接表的形式存储, 在一定程度上降低了算法时间复杂度, 以单节点出发的时间复杂度为 $O(n + e)$, 因此, 总体时间复杂度为 $O(Kn + e)$ 。

3.3 遗传算法全局优化过程

设 p 表示传播模型传播概率, $seedlist$ 表示包含 k 个种子节点的列表集合, SR 表示 3.2 节计算所得信息集, 遗传算法优化过程的适应性函数 CountSeed 描述如下:

输入 $seedlist, p, SR$

输出 这 k 个种子节点的综合影响概率

1. total_probability \leftarrow 0

2. for $t = 1$ to 3:

3. infect_list $\leftarrow \bigcup_{s \in seedlist} SR(s, t)$

4. count = Count(infect_list)

5. for user in count.keys():

6. number \leftarrow count[user]

7. probability $\leftarrow 1 - (1 - p^t)^{numbers}$

8. total_probability + = probability

9. return total_probability

该适应性函数 CountSeed 时间复杂度为 $O(n)$, 设 p 为传播模型各边传播概率, k 表示一个族群中种子个数, $popsiz$ 表示种群集合中种群数, $elite$ 表示种群集合中精英所占比例, $maxiter$ 表示最大迭代次数, $vatiprob$ 用来判定族群选择交叉还是变异, CIPIM 算法 GA 过程描述如下:

输入 $costfunc, k, popsiz, elite, maxiter, vatiprob, p$

输出 k 个种子组成的集合

1) 生成初始种群 $POP = \{P_1, P_2, \dots, P_{popsiz}\}$, 其中每个族群 P 都包含 k 个节点, 转到 2)。

2) 计算种群集每个族群综合影响概率, 将种群中 CountSeed 返回值较大者作为精英保留; 如果迭代次数达到 $maxiter$, 转到 5), 否则转到 3)。

3) 根据 $vatiprob$ 判断对保留下来的精英族群进行交叉操作还是变异操作, 并添加到新的种群集 POP 中, 转到 4)。

4) 如果 $len(POP')$ 小于 $popsiz$, 转到 3), 否则转到 2)。

5) 返回综合影响概率最高的种子集合。

4 实验结果与分析

4.1 实验数据集

为了全面分析 CIPIM 算法在不同规模网络环境下影响传播效果、可扩展性及性能, 本文选取规模不同的 2 个真实社交网络数据集 Wiki-Vote^[16] 和 Twitter^[18], 它们的统计特性如表 2 所示。

表 2 属性特征权重

数据集	节点数	边数	平均聚类系数
Wiki-Vote	7 115	103 689	0.140 9
Twitter	81 306	1 768 149	0.565 3

4.2 实验设置

为验证本文算法的性能, 选取当前较有代表性的算法进行比较。因为贪心算法在时间复杂度上较高, 即便是后续优化后的算法仍然需要较长、甚至数天的时间才能得到结果, 所以本文不与贪心算法进行比较, 而与 DegreeDiscount、PageRank、CCA 等算法进行比较。DegreeDiscount 算法是一种种子节点对邻居节点进行度折扣的启发式算法; PageRank 算法, 是 Google 用于标识网页重要性的算法, 本文中阻尼因子设置为 0.85; CCA(d) 算法是基于核数层次特征、消除重复影响的一种启发式算法, 原文中距离参数 d 为 2 时效果更好, 因此, 本文使用 CCA(2) 进行比较。本文算法 CIPIM 中备选种子比例 h 取为 10; 种群数一般设置为 20 ~ 100, 本文将种群数 $popsiz$ 设为 50; 最大迭代数一般取 100 ~ 500, 在本算法中因为对下一层精英选择过程做了特殊处理, 算法将加速收敛, 因此将最大迭代次数 $maxiter$ 设为 100; 同理, 精英策略 $elite$ 设为 0.2。

为了讨论传播概率对传播结果的影响, 排除实验结果的偶然性, 分别将影响传播概率设置为 0.01、0.03、0.06 进行蒙特卡洛模拟传播, 并对每次模拟传播进行 1 000 次实验, 取平均值作为传播结果。

实验硬件环境为 macOS, 内存 8 GB, 处理器 1.6 GHz Intel Core i5, 所有代码均使用 Python2.7.10 书写。

4.3 结果分析

影响最大化算法的评价通常从影响范围和时间效率 2 个方面衡量。

4.3.1 Wiki-Vote 数据集实验

Wiki-Vote 数据集是维基百科的一个投票数据,属于中型数据集。图 3 ~ 图 5 为各传播概率下的模拟实验平均被影响节点个数趋势。

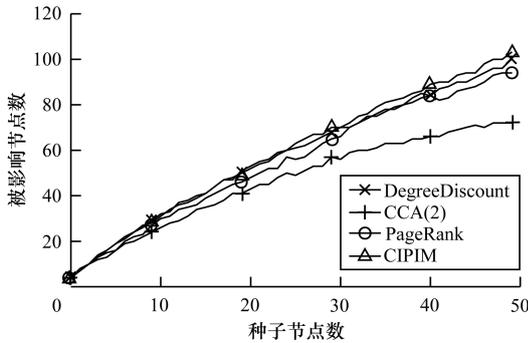


图 3 Wiki-Vote 上 $p = 0.01$ 时被影响节点数

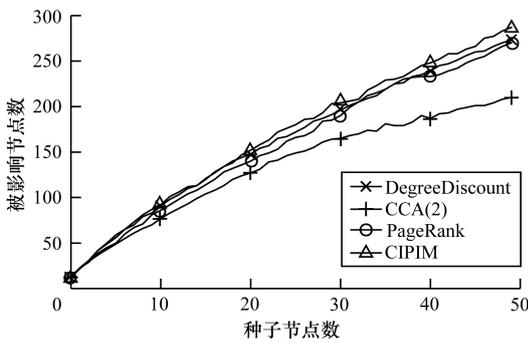


图 4 Wiki-Vote 上 $p = 0.03$ 时被影响节点数

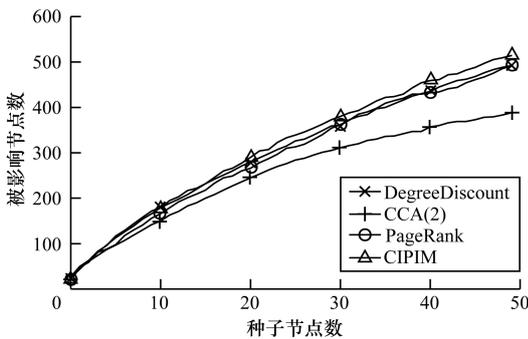


图 5 Wiki-Vote 上 $p = 0.06$ 时被影响节点数

从图 3 ~ 图 5 可以看出,在 $k < 10$ 时,各种算法在影响范围上结果较为接近,相差不大。但当 $k > 10$ 时,可以明显看出,CCA 算法比其他算法在影响范围上要稍稍逊色,且随着 k 值增大,差异也越来越大。CCA 算法为了减少算法运行时间,牺牲掉间接传播这一网络特性,导致其影响范围跟其他算法相比有些差距。CIPIM 算法在 PageRank 的算法基础上进行全局优化,其结果要好于单纯的 PageRank 算法。随着种子数 k 和传播概率 p 的变大,算法之间效果的差异也越来越大。表 3 为不同传播概率下各

算法的平均运行时间。

表 3 WikiVote 数据集下各算法平均运行时间 s

P	DegreeDiscount	PageRank	CCA(2)	CIPIM
0.01	0.582	1.174	0.368	1.379
0.03	0.619	1.174	0.371	1.656
0.06	0.653	1.174	0.385	1.974

由表 3 可以看出,CIPIM 算法在运行时间上比其他算法稍多一点,但时间差距并不算大,居于相同数量级下。由于种子节点数 k 对 CIPIM 算法有一定的影响,因此,随着 k 值的增大,算法所用时间也会随之稍稍增大。但对于贪心算法而言,CIPIM 算法用时已经相当低了。综上,CIPIM 算法在数据集 Wiki-Vote 上有良好表现,算法有效。

4.3.2 Twitter 数据集实验

Twitter 数据集属于大型社交网络数据集,虽然节点数不到 10 万个,但是却有着百万级别由关注关系形成的有向边。图 6 ~ 图 8 为各传播概率下的模拟实验被影响节点数趋势。

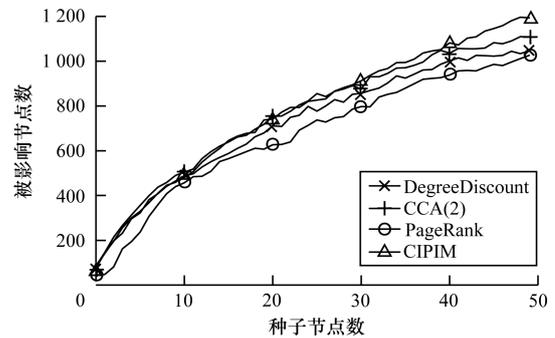


图 6 Twitter 上 $p = 0.01$ 时被影响节点数

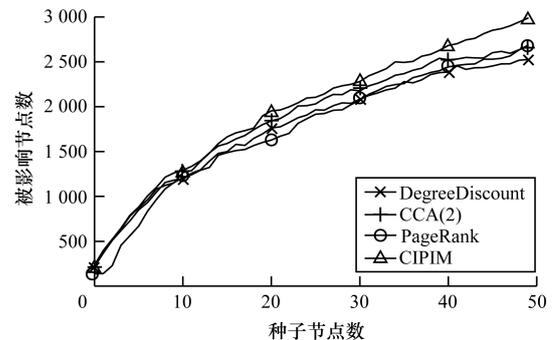


图 7 Twitter 上 $p = 0.03$ 时被影响节点数

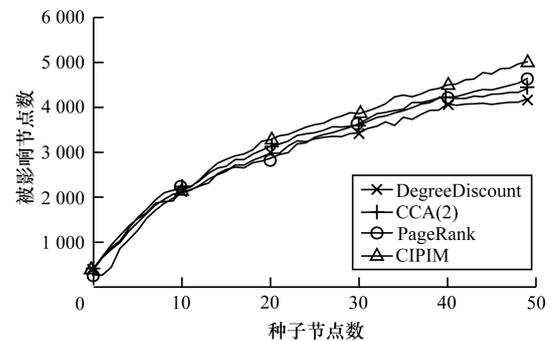


图 8 Twitter 上 $p = 0.06$ 时被影响节点数

从图6~图8可以看出,在较大型社交网络中,CIPIM算法表现优异。当 k 值较小时,除了PageRank算法表现一般,各算法相差不大。和Wiki-Vote数据集的运行结果相似,但是在 $k > 10$ 之后,算法之间的差异开始凸显,CCA算法在Twitter数据集上比DegreeDiscount算法要好上一点,说明在真正的社交网络上确实会有影响重叠的现象存在。CIPIM算法是从经PageRank算法排序后的节点中选择有较高影响力的节点作为备选节点,然后再通过潜在被激活节点的综合激活概率优化得到,所以在运行结果上,CIPIM算法始终高于PageRank算法。表4为不同传播概率下各算法的平均运行时间。

表4 Twitter数据集下各算法平均运行时间

p	DegreeDiscount	PageRank	CCA(2)	CIPIM
0.01	1.876	24.974	1.336	2.085
0.03	1.982	24.974	1.364	2.404
0.06	2.077	24.974	1.389	2.847

从表4可以看出,CIPIM算法在大型数据集上平均运行时间方面仍然表现良好,虽然比一些算法稍稍偏多,但是在影响范围覆盖度上弥补了这些不足。其中需要说明的是,在CIPIM算法中包含PageRank算法,但是只需计算一次,所以,在运行时间内没有包含在里面。

从以上实验结果表明,不管是在Wiki-Vote数据集还是Twitter数据集上,不管传播概率的取值如何,CIPIM算法均表现出较大优势。因为本文算法会在全局上进行优化,所以与直接从节点度或核出发的算法相比在平均时间上要略高,但在节点选取效果上要优于这些算法,且在时间效率上要远远好于贪心算法,能在影响覆盖度和运行时间上取得了较好的平衡。

5 结束语

本文提出一种基于合并影响概率的遗传算法,并利用该算法来解决影响最大化问题,通过缩减种子搜寻范围来减少工作量,使用遗传算法进行全局优化。实验结果表明,CIPIM算法改善了CCA算法在共同影响概率缺失方面的问题,同时与CCA算法、DegreeDiscount算法以及PageRank算法相比影响范围更广,时间复杂度远小于贪心算法。然而,本文算法仍存在不足之处,即各节点间的传播概率都是固定值,但在实际社交网络中并非如此。因此,在下一步工作中,将通过数据挖掘、机器学习等方法综合考虑多个社交网络,使其能够确定用户间不同的传播概率。

参考文献

- [1] DOMINGOS P, RICHARDSON M. Mining the network value of customers [C]//Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA: ACM Press, 2001: 57-66.
- [2] KEMPE D, KLEINBERG J, TARDOSE. Maximizing the spread of influence through a social network [C]//Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA: ACM Press, 2003: 137-146.
- [3] LESKOVEC J, KRAUSE A, GUESTRIN C, et al. Cost-effective outbreak detection in networks [C]//Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA: ACM Press, 2007: 420-429.
- [4] GOYAL A, LU W, LAKSHMANAN L V S. CELF++: optimizing the greedy algorithm for influence maximization in social networks [C]//Proceedings of International Conference on World Wide Web. New York, USA: ACM Press, 2011: 47-48.
- [5] CHEN Wei, WANG Yajun, YANG Siyu. Efficient influence maximization in social networks [C]//Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA: ACM Press, 2009: 199-208.
- [6] WASSERMAN S, FAUST K. Social network analysis [J]. Encyclopedia of Social Network Analysis & Mining, 2011, 22(Suppl 1): 109-127.
- [7] BRIN S, PAGE L. The anatomy of a large-scale hypertextual web search engine [J]. Computer Networks & Isdn Systems, 1998, 30(1-7): 107-117.
- [8] CHEN W, WANG C, WANG Y. Scalable influence maximization for prevalent viral marketing in large-scale social networks [C]//Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA: ACM Press, 2010: 1029-1038.
- [9] SONG Guojie, ZHOU Xiabing, WANG Yu, et al. Influence maximization on large-scale mobile social network: a divide-and-conquer method [J]. IEEE Transactions on Parallel & Distributed Systems, 2015, 26(5): 1379-1392.
- [10] ZHOU Chuan, ZHANG Peng, ZANG Wenyu, et al. On the upper bounds of spread for greedy algorithms in social network influence maximization [J]. IEEE Transactions on Knowledge & Data Engineering, 2015, 27(10): 1.
- [11] LI Ji, CAI Zhipeng, YAN Mingyuan, et al. Using crowdsourced data in location-based social networks to explore influence maximization [C]//Proceedings of IEEE Conference on Computer Communications. Washington D. C., USA: IEEE Press, 2016: 1-9.

- [7] CLARK J, KOPRINSKA I, POON J. A neural network based approach to automated e-mail classification [C]// Proceedings of 2003 IEEE/WIC International Conference on Web Intelligence. Washington D. C. , USA; IEEE Computer Society, 2003 : 702-705.
- [8] FORREST S, PERELSON A S, ALLEN L, et al. Self-nonsel discrimination in a computer [C]// Proceedings of 1994 IEEE Symposium on Security and Privacy. Washington D. C. , USA; IEEE Computer Society, 1994 : 202-212.
- [9] ODA T, WHITE T. Spam detection using an artificial immune system [EB/OL]. [2017-02-26]. <http://terri.zone12.com/doc/academic/crossroads/>.
- [10] MA W, TRAN D, SHARMA D. A novel spam email detection system based on negative selection [C]// Proceedings of the 4th International Conference on Computer Sciences and Convergence Information Technology. Washington D. C. , USA; IEEE Press, 2009 : 987-992.
- [11] MOHAMMAD A H, ZITAR R A. Application of genetic optimized artificial immune system and neural networks in spam detection [J]. Applied Soft Computing, 2011, 11 (4) : 3827-3845.
- [12] IDRIS I, SELAMAT A. Email spam detection using differential evolution negative selection algorithm [J]. International Journal of Digital Content Technology and Its Applications, 2013, 7 (15) : 15-20.
- [13] TAN Y, RUAN G. Uninterrupted approaches for spam detection based on SVM and AIS [J]. International Journal of Computational Intelligence and Pattern Recognition, 2014, 1 (1) : 1-26.
- [14] SIRISANYALAK B, SOMIT O. An artificial immunity-based spam detection system [C]// Proceedings of 2007 IEEE Congress on Evolutionary Computation. Washington D. C. , USA; IEEE Press, 2007 : 3392-3398.
- [15] SALTON G, BUCKLEY C. Term-weighting approaches in automatic text retrieval [J]. Information Processing and Management, 1988, 24 (5) : 513-523.
- [16] SPARCK J K. A statistical interpretation of term specificity and its application in retrieval [J]. Journal of Documentation, 1972, 28 (1) : 11-21.
- [17] ZUCHINI M H. Aplicações de mapas auto-organizáveis em mineração de dados e recuperação de informação [EB/OL]. [2017-02-25]. <http://viacodigo.com.br/pos/Zuchini,MarcioHenrique.pdf>.
- [18] BEZERRA G B, BARRA T V, FERREIRA H M. An immunological filter for spam [C]// Proceedings of International Conference on Artificial Immune Systems. Berlin, Germany; Springer, 2006 : 446-458.
- [19] DRUCKER H, WU D, VAPNIK V N. Support vector machines for spam categorization [J]. IEEE Transactions on Neural Networks, 1999, 10 (5) : 1048-1054.
- [20] JI Z, DASGUPTA D. Revisiting negative selection algorithms [J]. Evolutionary Computation, 2007, 15 (2) : 223-251.
- [21] 金章赞, 廖明宏, 肖刚. 否定选择算法综述 [J]. 通信学报, 2013, 34 (1) : 159-170.
- [22] 刘菊新, 徐从富. 基于多分类器组合模型的垃圾邮件过滤 [J]. 计算机工程, 2010, 36 (18) : 194-196.
- [23] 王祖辉, 姜维. 基于支持向量机的垃圾邮件过滤方法 [J]. 计算机工程, 2009, 35 (13) : 188-189.
- [24] ANDROUTSOPOULOS I, KOUTSIAS J, CHANDRINOS K V, et al. An experimental comparison of naive bayesian and keyword-based anti-spam filtering with personal e-mail messages [C]// Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, USA; ACM Press, 2000 : 160-167.

编辑 吴云芳

(上接第 193 页)

- [12] 曹玖新, 董丹, 徐顺, 等. 一种基于 k-核的社会网络影响最大化算法 [J]. 计算机学报, 2015, 38 (2) : 238-248.
- [13] MIAO Yu, WU Yang, WANG Wei, et al. UGGreedy: Influence maximization for user group in microblogging [J]. Chinese Journal of Electronics, 2016, 25 (2) : 241-248.
- [14] GRANOVETTER M. Threshold models of collective behavior [J]. American Journal of Sociology, 1978, 83 (6) : 1420-1443.
- [15] WATTS D J. A simple model of global cascades on random networks [J]. Proceedings of the National Academy of Sciences of the United States of America, 2002, 99 (9) : 5766-5771.
- [16] LESKOVEC J. Wikipedia vote network [EB/OL]. [2017-03-10]. <http://snap.stanford.edu/data/wiki-Vote.html>.
- [17] ELMACIOGLU E, LEE D. On six degrees of separation in DBLP-DB and more [J]. Acm Sigmod Record, 2005, 34 (2) : 33-40.
- [18] LESKOVEC J. Social corcles; twitter [EB/OL]. [2017-03-10]. <http://snap.stanford.edu/data/egonets-Twitter.html>.

编辑 刘冰