•信息工程•

DOI:10.15961/j.jsuese.201700728

基于线性阈值模型的动态社交网络影响最大化算法

朱敬华,李亚琼,王亚珂,杨　艳*

(黑龙江大学 计算机科学与技术学院,黑龙江 哈尔滨 150080)

摘　要:针对随时间进化的动态社交网络展开影响最大化问题的研究,目标是基于线性阈值传播模型,挖掘影响力最大的k个种子用户,从种子用户发起传播,最大化影响传播范围。提出一种基于线性阈值模型的动态社交网络影响最大化算法(linear threshold dynamic influence maximization,LTDIM)。首先,给出动态社交网络影响最大化问题的形式化定义,提出利用活边路径获取初始种集的方法;然后,分析网络的各种拓扑变化,提出种集的增量式更新方法;最后,基于节点度和影响力增量提出DP(degree pruning)和IIP(influence increment pruning)剪枝策略进一步提高时间效率。实验使用4个真实的社交网络数据,考察在8个网络快照上算法的运行时间和影响传播范围。实验结果表明,本文算法的影响传播范围接近于静态启发式算法,运行时间大幅度减少,验证了算法的时间高效性和可扩展性。

关键词:动态社会网;影响最大化;线性阈值模型;剪枝策略

中图分类号:TP311　　**文献标志码**:A　　**文章编号**:2096-3246(2019)01-0181-08

Influence Maximization Algorithm for Dynamic Social Networks Based on Linear Threshold Model

ZHU Jinghua, *LI Yaqiong*, *WANG Yake*, *YANG Yan**

(School of Computer Sci. and Technol., Heilongjiang Univ., Harbin 150080, China)

Abstract: In order to solve the influence maximization problem in evolving social network, a dynamic influence maximization algorithm based on the linear threshold model was proposed in this paper. The goal of influence maximization was to mine out the top k most influential seed users and maximize the spread of influence through them. An algorithm called LTDIM was proposed based on the linear threshold model. Specially, firstly, the formal definition of dynamic influence maximization problem was given and the initial seeding method based on alive edge path was proposed. Then, according to the analysis of various network topology changes, an incremental seeds updating algorithm was presented. Finally, to further improve the time efficiency, two pruning strategies DP (degree pruning) and IIP (influence increment pruning) based on nodes degree and influence increment were devised. Experiments on the eight network snapshots of four real social networks evaluated the algorithm performance in terms of running time and influence spread. Experimental results demonstrated that compared with the state-of-the-art static heuristic algorithms, the algorithms proposed in the paper can achieve a great deal of speedup in running time while maintaining matching performance in terms of influence spread.

Key words: dynamic social networks; influence maximization; linear threshold model; pruning strategy

近年来,在线社交网络纷纷兴起,如新浪微博、Facebook、Twitter等。社交网络不仅是人们交流信息,结识朋友的重要媒介,也是重要的商业平台,如商家可以在社交网络中选取一小部分有影响力的用户,给这些用户免费提供产品,试用者通过社交网络向朋友或家人推荐产品,以“口口相传”的方式实现产品的推广。因此,社会网络中的影响最大化问题逐渐引起研究者的广泛关注[1]。影响最大化问题旨在网络中选取k个最具影响力的种子用户,由这些用户发起信息传播,目标是将影响扩散到整个网络[2]。例如,Kempe等[3]利用离散优化方法解决影响最大化问题,提出一种具有近似精度保证的贪婪算法。但是,该贪

收稿日期:2017 – 09 – 08

基金项目:国家自然科学基金资助项目(61100048;61370222);黑龙江省自然科学基金资助项目(F2016034;F2018028)

作者简介:朱敬华(1976—),女,教授,博士,硕士生导师. 研究方向:社会网;数据挖掘;无线传感器网络. E-mail:zhujinghua@hlju.edu.cn

* 通信联系人 E-mail:yangyan@hlju.edu.cn

网络出版时间:2019 – 01 – 16 11 : 07 : 29　　**网络出版地址**:http://kns.cnki.net/kcms/detail/51.1773.TB.20190115.1408.001.html

婪算法的时间开销相当大，因此，后续研究者[4–6]研究各种近似算法和启发式算法，目的是降低影响最大化问题的时间开销。

然而，上述算法都基于一个相同的假设，即静态网络。现实的社交网络并非是一成不变的，随时间持续变化。例如，新用户注册、旧用户注销、关系加入或删除等原因都会引起网络拓扑的变化。新浪微博每天大约有3万个联系加入或删除。社交网络的快速变化给影响最大化带来新的挑战，在社交网络持续发生变化时，如何快速发现新的最有影响力的用户，这是一个有待研究的问题。网络的拓扑变化引起种子集合和影响传播范围的变化，现有的静态算法不能够捕捉并处理这些变化，而且更新种集的开销巨大。

目前，已有关于动态社交网络影响最大化问题的研究。例如：Gayraud等[7]针对时间进化的社会网影响传播问题，提出EIC模型和ELT模型，通过实验证明了在静态网络图上计算的影响范围往往超过实际影响范围很多。Zhuang等[8]提出基于探测子集的近似算法MaxG，根据探测节点构成的局部网络计算种集，作为全局的近似解。MaxG虽然能够解决动态社会网络的影响最大化问题，但在选择探测节点时需要对全网络中所有节点进行计算，开销相当大。Tong等[9]将传统独立及联模型扩展为动态独立级联模型DIC，提出A-Greedy贪心算法和H-Greedy启发式算法。Liu等[10]基于IC模型提出增量式种集更新算法，描述了网络进化的过程，通过捕获局部网络拓扑变化进行种集更新，有效地降低了计算开销。Bao等[11]基于多臂老虎机模型提出随机化算法RSB，解决网络动态变化时的影响最大化问题。上述研究虽然能够解决动态网络的影响最大化问题，但是，都基于独立级联模型，而本研究则基于线性阈值模型提出一种动态社交网络影响最大化算法（linear threadhold dynamic influence maximization, LTDIM）。

作者提出的LTDIM算法分为两个阶段：阶段1是初始种集获取，阶段2是增量式种集更新。阶段1，遍历获得简单路径集合，根据该路径集合计算影响力范围，得到初始种集并为后续的种集更新创造初始条件。阶段2，采用影响局部化思想捕获拓扑变化引起的影响变化，在动态进化中快速确定最有影响力节点。为进一步降低时间开销，作者还提出两种剪枝策略，即影响力增值剪枝策略和度剪枝策略。在网络变化后，若已知种子的影响范围增大，则采用影响力增值剪枝策略保留变化幅度大的节点；若种子的影响范围减小，则采用度剪枝策略选择度数或度增长率排名靠前的节点，调整候选集，减少计算量。在真实社交网络数据集上进行验证实验，结果表明，和静态算法相比，本文算法能够获得相匹配的影响传播范围，但是算法的运行时间却大幅度减少，适用于大规模动态社交网络。

1 预备知识

1.1 社交网络模型

用加权有向图$G=(V,E,W)$表示社交网络，其中：点集V代表用户；边集$E\subseteq V\times V$代表用户之间的联系；$W:V\times V\to[0,1]$是边上的加权函数，表示用户之间的影响概率。若边$(u,v)\notin E$则$w(u,v)=0$，并且$\sum_{u\in V}w(u,v)\le 1$。

根据用户行为的发生情况，每个节点通常有两种状态，即激活状态或者非激活状态，且同一时刻只能处于其中一种状态。当节点从非激活状态变为激活状态后，该节点会尝试激活其非激活状态的邻居节点。在传播过程中，节点的状态只能从非激活状态变为激活状态。如果一个节点周围有越来越多的邻居节点变为激活状态，则该节点被激活的可能性也越来越大。

1.2 影响传播模型

影响传播模型是影响力在社会网络中的传播方式和机制，广泛研究的影响模型有独立级联模型（independent cascade model, IC）和线性阈值模型（linear threshold model, LT）。本文算法基于线性阈值模型，下面详细介绍LT模型的机制。

LT模型主要关注影响力传播过程中的阈值行为，即影响力在传播过程中所具有的累积效应。当一个激活状态的节点尝试激活其非激活状态的邻居节点时，尝试失败节点在该次激活过程中的影响力会被累积，并对后面其他节点对该非激活节点的激活行为产生贡献。也就是说，节点是否被成功激活由多个已激活的前驱节点的影响权重共同决定，因此，激活行为并不独立，满足$\sum_{u\in pre(v)}w(u,v)\le 1$，其中，$pre(v)$为节点$v$的前驱节点集合。LT模型为每个节点赋予阈值$\theta_v\in[0,1]$，阈值越大，越不容易被激活。初始$t$=0时刻，只有种子集合$S$中的节点是激活状态。假设节点$x$在$t$–1时刻变为激活状态，则$x$将以$w(x,v)$的概率影响其非激活状态的后继节点$v$。节点$v$在$t$时刻被激活的条件是$\sum_{x\in pre(v)}w(x,v)\ge\theta_v$。传播过程直到网络中再没有节点可以被激活时停止。用$\sigma(S)$表示种集S的影响传播范围。

1.3 活边模型

Kempe等[3]提出了活边模型：节点v以概率$w(u,v)$

保留入边，以$1-\sum_{u} w(u,v)$的概率丢弃入边。被保留的入边称为活边，被丢弃的入边称为阻塞边。设G_{L}是G的只包含所有活边的生成子图，称为活边图。在G_{L}中，若节点u可达节点v，则一定存在从u到v的活边路径。Kempe证明了给定种集S，线性阈值模型和活边模型的影响传播范围相同，因此，可以用式（1）计算种集S的影响范围：

$$\sigma(S)=\sum_{G_{\mathrm{L}}} pro(G_{\mathrm{L}})\cdot\sigma_{G_{\mathrm{L}}}(S) \tag{1}$$

式中，$pro(G_{\mathrm{L}})$为活边图的出现概率，$\sigma_{G_{\mathrm{L}}}(S)$为$G_{\mathrm{L}}$中从种集$S$出发可达节点的期望。

$I_{G_{\mathrm{L}}}(S,v)$是一个指示函数，如果在活边图G_{L}中存在从S中的节点到达节点v的活边路径，则其函数值为1；否则，为0。因此，种集S在活边图G_{L}上的影响范围可以用式（2）计算得到：

$$\sigma_{G_{\mathrm{L}}}(S)=\sum_{v\in V} I_{G_{\mathrm{L}}}(S,v) \tag{2}$$

式中，V是图的顶点集合。将式（2）代入到式（1）中得到：

$$\sigma(S)=\sum_{v\in V}\sum_{G_{\mathrm{L}}} pro(G_{\mathrm{L}})\cdot I_{G_{\mathrm{L}}}(S,v)=\sum_{v\in V}\sigma(S,v) \tag{3}$$

式中，$\sigma(S,v)$为S是种集时节点v的激活概率。

2　动态社交网络影响最大化问题

2.1　传统影响最大化问题

给定社交网络图$G=(V,E,W)$和种集大小k，影响最大化问题旨在挖掘含有k个节点的具有最大影响传播范围的种子集合S^{*}，该问题的目标函数可以用式（4）描述：

$$S^{*}=\underset{S\subseteq V,|S|=k}{\arg\max}\ \sigma(S) \tag{4}$$

Kempe等[3]证明了影响最大化问题在线性阈值模型下是NP难问题，提出了解决该问题的贪心算法。节点v在种集为S时的边际影响MI（marginal influence）定义为$MI(v|S)=\sigma(S\cup\{v\})-\sigma(S)$。用蒙特卡洛模拟得到节点的边际影响，边际影响$MI$函数具有单调性和子模性，能够保证贪心算法获得（1–1/e）的近似比。

2.2　动态影响最大化问题

给定网络快照序列$G=\{G^{1},G^{2},\cdots,G^{n}\}$，其中，$G^{t}=(V^{t},E^{t},w^{t})(t=0,1,\cdots,n)$是社会网在$t$时刻的拓扑快照。$\Delta G^{t}=(\Delta V^{t},\Delta E^{t},\Delta w^{t})$是$G^{t}$在$t$时刻的拓扑变化量，则$t$+1时刻的网络拓扑$G^{t+1}=G^{t}\cup\Delta G^{t}$。动态社交网络连续影响最大化问题的目标函数可以表示为：

$$S^{t^{*}}=\underset{S^{t}\subseteq V,|S^{t}|=k}{\arg\max}\ \sigma(S^{t})\ \ (t=0,1,\cdots) \tag{5}$$

式中，S^{t}为t时刻的候选种子集合。

3　动态社交网络影响最大化算法（LTDIM）

如果将各时刻的网络快照输入到传统影响最大化算法中求解，则算法的时间开销巨大，不能适应大规模社交网络的实时传播。作者提出动态影响最大化算法（LTDIM），以增量的方式连续捕获网络的动态变化，用较小的代价更新种集。LTDIM算法分为两个阶段：阶段1是计算初始种集，阶段2是增量式更新初始种集。

3.1　初始种集获取

动态社会网络是持续变化的，在离散时间步观察社会网的变化，网络拓扑在各个时刻是静态图。假设时刻t=0的社会网为初始图，则根据式（1）可以计算种集S在初始图G的传播范围。

定　义　设$P=\langle v_1,v_2,\cdots,v_m\rangle$是一条简单路径，$(v_i,v_j)\in P$表示边$(v_i,v_j)$在路径$P$上。路径$P$是活边路径的概率计算如下：

$$pro(P)=\prod_{(v_i,v_j)\in P} w(v_i,v_j) \tag{6}$$

则节点u对节点v的影响力为：

$$\sigma(u,v)=\sum_{P\in Path_{u,v}} pro(P) \tag{7}$$

式中，$Path_{u,v}$为节点u到节点v的所有路径集合。由此，节点u的影响范围表示如下：

$$\sigma(u)=\sum_{v\in V}\sigma(u,v) \tag{8}$$

集合S的影响传播范围是该集合中所有节点的影响范围之和，即

$$\sigma(S)=\sum_{u\in S}\sigma^{V-S+u}(u) \tag{9}$$

节点v的边际影响$MI(v)=\sigma(S\cup\{v\})-\sigma(S)$。根据式（9），计算时要为每一个$x\in S\cup\{v\}$在对应的导出子图$G[V-S-v+x]$中计算$\sigma^{V-S-v+x}(S\cup\{v\})$，除节点$x$外，每加入一个新节点，即使拓扑变化甚微，都要重新计算节点的影响力。Goyal等[5]证明了对于变化微小的子图，节点的影响力不会受到大的影响，因此，$\sigma(S\cup\{v\})$的计算公式推导如下：

$$\begin{aligned}\sigma(S\cup\{v\})&=\sum_{x\in S\cup\{v\}}\sigma^{V-S-v+x}(x)=\\&\sigma^{V-S}(v)+\sum_{x\in S}\sigma^{V-S-v+x}(x)=\\&\sigma^{V-S}(v)+\sigma^{V-v}(S)\end{aligned} \tag{10}$$

为了计算初始种集，利用路径搜索函数GetAllPath获得到达给定节点的所有简单路径，并利用这些路径计算节点的影响力。GetAllPath利用回溯思想，从节点u出发沿出邻节点深度遍历，若当前路径$P=\langle u,\cdots,x,v\rangle$的影响概率小于$\eta$，则回溯到路径的上一个节点$x$，检查$x$是否还有其他出邻节点，沿其他出邻节点继续深度遍历；反之，则继续向前回溯。遍历时记录路径，并累加影响概率得到节点的影响值，回溯到初始路径的起始节点u算法结束。

由于路径的概率随路径长度增加而减小，被拓扑变化影响的节点仅局限于小范围。为了能够准确、快速确定受拓扑变化影响的节点，为每个节点$u\in V$维护$IN(u)$集合，存储阈值η内对u产生影响的节点。通过UpdateIN(u)函数更新$IN(u)$集合，思想如下：对于节点$v\in V$，若v的路径集合发生变化，则遍历其路径集合$Path(v)$中的每一条路径；如果$IN(u)$包含节点u，则将节点v加入$IN(u)$集合；如果$IN(u)$已经包含节点v，不重复添加。由此得到图中所有能够到达节点u的节点集合。利用IN集合，网络变化发生后，只需重新计算受影响的节点的影响力，而不需要全网络计算，从而减少时间开销。

下面介绍种集初始化（Init_Seed）算法。先调用GetAllPath为图中每个节点计算初始影响值，并根据节点的$Path$集合更新IN集合；在每次迭代中，根据贪心思想，选择边际影响增益最大的节点加入种集，直至种集达到设定大小。Init_Seed算法的伪代码如下：

算法1 种集初始化算法（Init_Seed）

输入：社交网络G，种集大小k，控制阈值η。

输出：种子集合S。

```
1.  FOR 任意u ∈ G(V) DO
2.      σ(u)=GetAllPath(u);
3.      UpdateIN(u); 将u加入到CELF队列
4.  END
5.  S ← ∅, spd←0;
6.  WHILE |S| < k
7.      取CELF队列的队首节点u;
8.      FOR s ∈ S //计算σ^{V−u}(S)
9.          FOR 包含节点u的路径P ∈ Path(s)
10.             σ^{V−u}(s) −= pro(P);
11.         END
12.         spd+ = σ^{V−u}(s);
13.     END
14.     FOR P ∈ Path[u]且P∩S ≠ ∅ //计算σ^{V−S}(u)
15.         σ^{V−S}(u) −= pro(P);
16.     END
17.     根据式(10)计算σ(S+u);
18.     MI(u)=σ(S+u)−spd;
19.     IF MI(u)大于CELF队列首节点的影响值
20.         u加入种集S; 删除节点u;
21.     ELSE
22.         将u插入CELF队列;
23.  RETURN S
```

算法1第8～12行表示，已经得到了节点的路径集合，在$V-u$的生成子图上计算种子的影响值时，若从种子出发包含u的路径无效，减去对应路径的影响值。第14和15行是为节点u计算在导出子图$G[V-S]$上的影响值。第17和18行是计算节点的边际影响。第19～22行表示：如果节点u的$MI(u)$大于队列首节点的影响值，直接将其选为种子节点；否则，将节点再次插入CELF队列。循环直至选出k个种子。

3.2 增量式种集更新

社会网在离散时间的变化用一系列快照图描述，任意时刻$t>0$，t时刻的图是$t-1$时刻变化后的新图。网络的拓扑变化可以分为6种：添加边，删除边，添加节点，删除节点，增加权重，减小权重。重点描述添加边和删除边算法，其他操作类似。

添加边的伪代码如下：

算法2 添加边算法

输入：新边(u, v, w)。

输出：节点的影响变化值$\Delta\sigma(x)$，$x\in C$。

```
1.  FOR 每一个节点x ∈ C DO
2.      Δσ(x)=0;
3.      FOR 路径P_{x,u} ∈ Path(x) DO
4.          IF pro(P_{x,u})·w(u,v) > η且P_{x,v}无重复节点
5.              添加P_{x,v}到temp;
6.              IF P_{x,v}与S交集为∅
7.                  Δσ(x)+ = pro(P_{x,v});
8.      IF temp不为∅
9.          FOR P_{x,v}属于temp DO
10.             FOR P_v, 属于Path(v) DO
11.                 IF pro(P_{x,v})·pro(P_v) > η, (u,v) ∈
                        P_x且P_x无重复节点
12.                     添加P_x到temp;
13.                     IF P_x与S交集为∅
14.                         Δσ(x)+ = pro(P_x)
15.     将temp中的路径全部复制到Path(x);
16.     调用UpdateIN(x)更新IN集合;
17.  RETURN Δσ(x)
```

在算法2中，w为新边(u,v)上的概率，集合C保存受添加边所影响的节点，$temp$暂存新增路径，$\Delta\sigma(x)$为节点u的影响变化值，$P_{x,u}$表示由从x到u的路径，P_v表示从节点v出发的路径。算法2为每个节点计算$\Delta\sigma(x)$，

根据最新$Path$集合更新节点的IN集合。第3～7行表示: 找到从x到u的路径, 连接新边(u,v); 如果新路径的影响概率大于控制阈值, 则存入$temp$中; 如果新路径不包含种子节点, 则将路影响概率累加至$\Delta\sigma(x)$。第8～14行表示: 如果$temp$为空, 说明节点过新边的路径或者影响概率太小, 或者路径有环, 算法结束; 否则, $temp$中的新路径再尝试连接节点v的$Path$集合中的路径。第15行表示将新增路径加入到$Path(x)$中, 第16行表示根据x的新$Path$集合更新节点的IN集合。

下面分析添加边算法的时间复杂度。假设节点的路径集合中路径数最多为$P_{\max}$, $temp$中暂存路径最多为$O(P_{\max})$, 节点的原路径都可以延伸至新边, 那么, 第3～7行的时间复杂度为$O(P_{\max})$, 第8～15行的时间复杂度为$O(P_{\max}\cdot P_{\max})$, 第16行的时间复杂度为$O(P_{\max})$, 所以, 算法的时间复杂度为$O(|C|\cdot P_{\max}^2)$。

删除边的伪代码如下:

算法3　删除边算法

输入: 删边(u, v)。

输出: 节点的影响变化值$\Delta\sigma(x), x\in C$。

1. FOR $x\in C$ DO
2. $\Delta\sigma(x)=0$;
3. FOR 路径$P_x\in Path(x)$且$(u, v)\in P_x$
4. $\Delta\sigma(x)-=pro(P_x)$; // 删除路径$P_x$
5. 调用UpdateIN(x);
6. RETURN $\Delta\sigma(x)$

算法3代码第1～4行表示确定受影响节点x的所有无效路径, 得到节点x的影响变化值$\Delta\sigma(x)$, 删除所有无效路径。第5行表示根据节点x的新路径集合$Path$更新节点的IN集合。若节点x的$Path$集合中的路径数为$|Path(x)|$, 算法第3、4、5行的时间复杂性都为$O(P_{\max})$, 所以, 算法时间复杂性为$O(2\cdot|C|\cdot P_{\max})$。

增量式种集更新算法的伪代码如下:

算法4　增量式更新种子算法

输入: t–1时的网络图G^{t-1}, t–1时的种集S^{t-1}, t时的网络图G^t, 种集大小 k。

输出: t时的种集S^t。

1. $S^t\leftarrow\varnothing$
2. WHILE $|S^t|<k$
3. FOR 从G^{t-1}到G^t的每一个改变c DO
4. 得到相应的受影响节点集合C;
5. FOR节点$u\in C$ DO
6. 调用相应的算法计算节点的$\Delta\sigma(u)$;
7. 添加u到S^C候选集;
8. FOR u属于S^C do
9. $MI(u)=\sigma(S^t\cup\{v\})-\sigma(S^t)$;
10. $S^t=S^t\cup\text{argmax}\ MI(u)$;
11. RETURN S^t

算法4中: S^C是非种子节点的候选集, S^t是t时刻的种集; 在选择种子节点时, 仅从S^C中选取, 缩小选取节点的范围, 减少计算量。第1行表示初始化。第3～7行表示, 根据图中发生的变化调用相应的6种操作算法, 计算节点的影响变化值, 筛选候选节点。第8～10行表示, 计算边际收益贪心选择种子。

3.3　优化算法(Opt_LTDIM)

社会网规模庞大, 节点数量众多, 而种子集合相对较小, 故在影响最大化算法的迭代计算过程中, 对于数量极大却不可能成为种子的节点, 计算其影响值, 浪费大量时间。虽然在提出的LTDIM算法中只对影响值发生改变的节点重新评估影响范围, 但变化量很小的节点仍然不足以成为有影响力的节点, 因此, 为进一步减少计算开销, 提出两个剪枝策略, 对候选集进一步筛选。

1)影响力增值剪枝策略(influence increment pruning)。在第i次迭代中, 图G^{t-1}中的种集S^{t-1}的影响变化值为正, 若节点v的影响变化值大于S^{t-1}中任一种子节点的影响变化值, 则保留为候选节点; 反之, 则筛选掉该节点。大多数情况下, 图G^{t-1}中的最具影响力的节点会吸引大量的节点并建立新的联系, 因此, 影响值会增加, 影响变化值为正。若非种子节点v的影响变化值小于S^{t-1}中任一种子节点的影响变化值, 同时由于在图G^{t-1}中, 节点v的影响值也小于S^{t-1}中种子节点的影响值, 所以节点v在图G^t中的影响值一定小于S^{t-1}中任一节点的影响值, 这样的节点不可能成为G^t中的种子节点。社会网络满足优先连接规则, 影响力增值剪枝策略可以剪枝大量节点, 减少计算量。

2)度剪枝策略(degree pruning)。由于社会网络中存在优先连接规则, 新增边以更大地概率优先连接度数大的节点, 社会网络发生拓扑变化时, 度数大的节点所受到的变化幅度会更大。若图G^{t-1}中的种子集合S^{t-1}的影响变化值为负, 即S^{t-1}的影响范围减少了, 则除了参照第1个剪枝策略, 保留节点还应满足以下两个条件之一: ①节点的度排名在G^t所有节点中位于前5%; ②节点的度增长率在图G^t所有节点中排名前5%。

将Inc_Seed算法的第7行改为: 根据剪枝策略选择u加入S^C候选集, 能够进一步减少候选集合大小, 在保证较高准确性的前提下, 提高算法时间效率。加入剪枝策略的LTDIM算法称为Opt-LTDIM算法。

4　实验和评估

4.1　实验设置

选用4个数据集测试不同算法在动态社交网络

中的性能。第1个数据集NetHEPT是一个学术合作网络，从arXiv电子版中“高能物理理论”部分抽取的文献引用网络。第2个数据集Facebook，数据来自于Facebook部分用户之间的好友关系网络。第3个数据集Flixster，它是一家社交电影网站。第4个数据集Flickr，数据来源是雅虎旗下的Flickr社会网。数据采集包括2015年1月—3月以及2015年12月—2016年4月，共历时8个月，每个月分别在4个数据集上进行快照记录，每个数据集都包含有8个网络快照。

表1、2列出了数据集中节点和边的初始个数、最终个数及节点增长率和边增长率。由表1、2可以看出，真实的社交网随时间推进快速变化。NetHEPT数据集节点数和边数都是最小的，Flick数据集规模最大。

表 1　节点信息

Tab. 1　Node information

数据集	节点信息		
	初始个数	最终个数	增长率/%
NetHEPT	15 634	18 557	18.7
Facebook	59 736	83 983	40.3
Flixster	99 825	147 328	48.5
Flickr	771 738	1 037 995	34.5

表 2　边信息

Tab. 2　Edge information

数据集	连接信息		
	初始个数	结束个数	增长率/%
NetHEPT	62 836	89 415	42.3
Facebook	576 653	994 149	72.4
Flixster	978 265	1 811 249	85.2
Flickr	4 938 687	7 106 122	43.8

通过实验对比动态网络影响最大化算法的性能，评价指标有运行时间和影响范围。其中：运行时间是发现最有影响力的k个节点的时间，体现了算法的执行效率；影响范围是种集在社交网络中最终影响的节点数量的期望值，体现了算法的精准度。

节点的阈值用随机函数获取。边的影响概率计算如下：节点v的各入边影响概率是节点v的入度的倒数，即$1/d_v$。

4.2　实验结果分析

将提出的LTDIM算法和Opt-LTDIM算法同两个静态网络基于线性阈值模型的影响最大化算法LDAG[4]和SIMPATH[5]进行比较。算法运行时间评价标准为在数据集中发现最有影响力的50个节点所用时间。图1(a)～(d)为本文算法和对比算法在4个不同数据集(NetHEPT、Facebook、Flixster、Flickr)的运行时间对比。其中，横轴是社会网络的8个快照，纵轴是算法的运行时间。

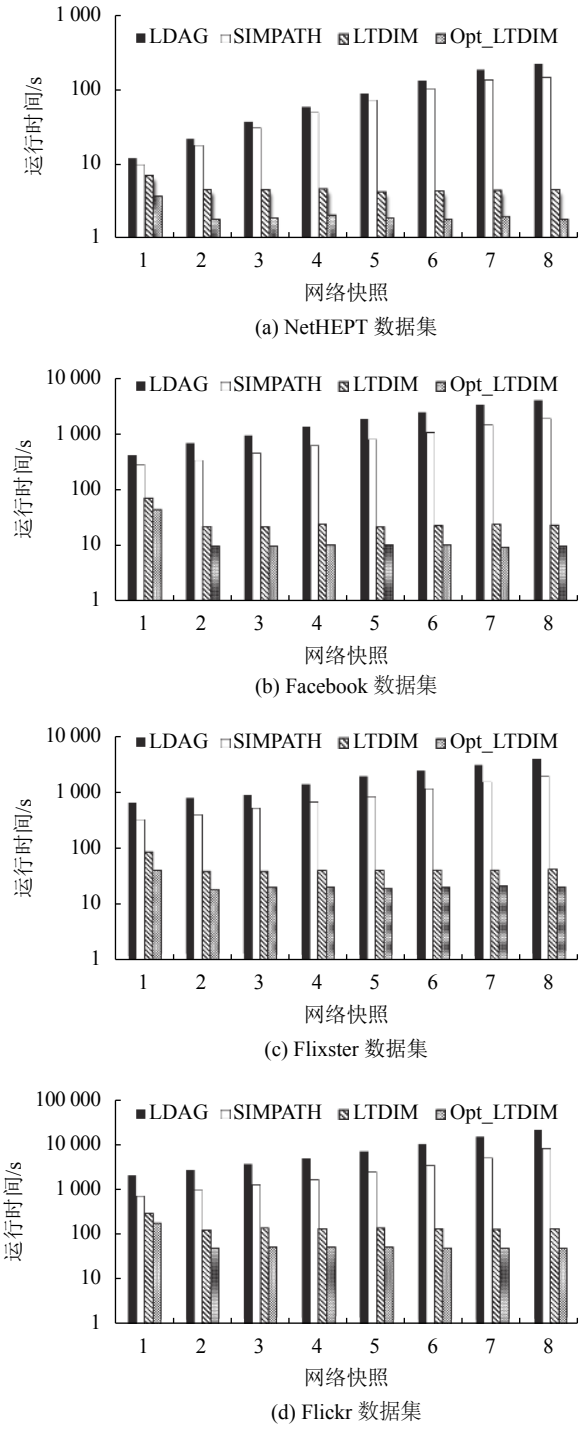


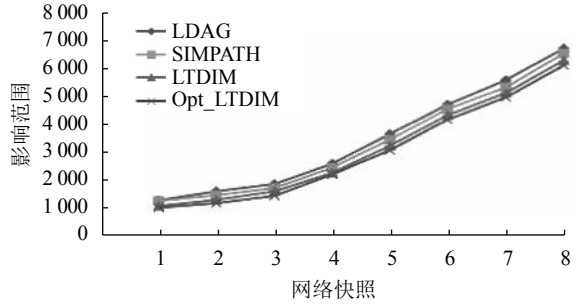
图 1　4个真实数据集上算法的运行时间

Fig. 1　Running time of different algorithms on four real datasets

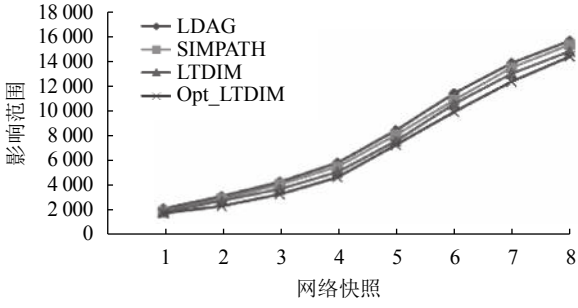
从图1可以看到：本文提出的LTDIM和Opt-LTDIM算法在4个数据集上的运行时间比静态算法LDAG和SIMPATH算法平均快6.3倍、13.6倍、17.4倍和31倍。本文动态算法LTDIM和Opt-LTDIM都在第1个网络快照调用算法1获得初始种集，因此运行时间和静态算法相近。后续7个网络快照时，动态算法只调用增量式更新算法4，运行时间小于静态算法。图1(d)

Flickr数据集属于大型网络，本文动态算法LTDIM和Opt-LTDIM的执行时间比前3个较小数据集的时间长，但仍然比静态算法执行时间短。Opt-LTDIM算法由于采用了剪枝策略，比LTDIM算法时间效率更高，在Facebook数据集上Opt-LTDIM算法比LTDIM算法平均快1倍，而在Flixster和Flickr数据集上平均要快出2倍多，验证了提出的优化策略的有效性。

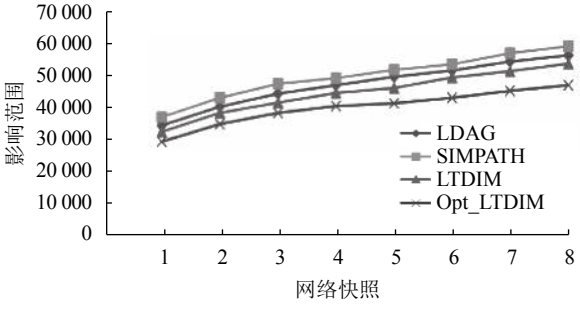
图2(a)～(d)给出了当种子集合大小为50时本文算法和对比算法在4个数据集上运行的影响传播范围。其中，对比算法LDAG[4]基于节点的局部有向无环图计算节点影响值，SIMPATH[5]基于简单路径估计影响值。



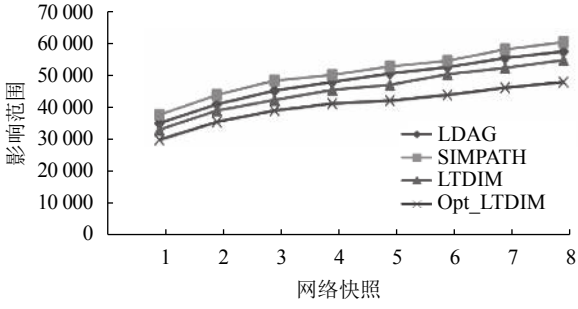
(a) NetHEPT 数据集



(b) Facebook 数据集



(c) Flixster 数据集



(d) Flickr 数据集

图 2　4个真实数据集上算法的影响范围

Fig. 2　Influence spread of different algorithms on four real datasets

如图2(a)、(b)所示，在NetHEPT和Facebook数据集上LTDIM和Opt-LTDIM与静态算法的影响范围非常接近。如图2(c)、(d)所示，在Flixster和Flickr这样较大数据集上，静态算法LDAG与SIMPATH的影响范围较动态算法略有优势，这是由于LDAG算法与SIMPATH算法在每次迭代中都在全网范围内重新计算节点的影响力，而本文提出的动态算法只计算局部受影响的节点的影响力，因此，影响范围略低于静态算法。但动态算法在快速提升执行速度的同时，算法精度接近于静态算法。

由2(c)～(d)可知：Facebook，Flixster和Flickr数据集上，SIMPATH算法优于LDAG算法，LTDIM算法优于Opt-DIM算法。这是因为基于节点影响改变值和优化策略在缩减候选节点规模的同时，也会带来一定的误差。LTDIM和Opt-LTDIM分别在Facebook、Flixster和Flickr数据集上比静态算法SIMPATH的影响范围平均低4.1%、5.2%和5.9%。虽然，提出的动态影响最大化算法的精确度低于静态算法，但是，从图1(a)～(d)的运行时间可以看出动态算法更适合大规模社交网的影响最大化问题。

5　结　论

提出了动态网络影响最大化算法，该算法包括初始种集获取和增量式种集更新两个阶段。通过遍历全网得到所有节点的路径集合，并根据路径集合快速计算节点的边际影响值；更新算法确定局部受影响的网络子图，提高了算法的时间效率。还提出了两种剪枝策略以缩小候选集，进一步提高时间效率。通过真实社会网的实验验证了提出算法的性能优势，当网络随时间频繁动态变化时，本文的算法能够以较小的计算成本在较短时间内获得最大影响范围的传播种集，提高市场营销和信息传播的实时性和高效性。

社会网用户之间的关系是不断变化的，边概率也是非确定的，下一步将研究社会网中边概率的分布，基于边概率分布变化研究动态网络影响最大化算法。

参考文献：

[1] Zhou Shengfu,Yue Kun,Fang Qiyun,et al.An efficient algorithm for influence maximization under linear threshold

model[C]//Proceedings of the 26th Control and Decision Conference.Changsha:IEEE,2014:5352-5357.

[2] Guo Jing,Zhang Peng,Fang Binxing.Personalized key propagating users mining based on LT model[J].Chinese Journal of Computers,2014,37(4):809–818.[郭静,张鹏,方滨兴,等.基于LT模型的个性化关键传播用户挖掘[J].计算机学报,2014,37(4):809–818.]

[3] Kempe D,Kleinberg J,Tardos É.Maximizing the spread of influence through a social network[C]//Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.NewYork:ACM,2003:137-146.

[4] Chen Wei,Yuan Yifei,Zhang Li.Scalable influence maximization in social networks under the linear threshold model[C]//Proceedings of the 2010 IEEE International Conference on Data Mining.Sydney:IEEE,2011:88-97.

[5] Goyal A,Lu Wei,Lakshmanan L V S.SIMPATH:An efficient algorithm for influence maximization under the linear threshold model[C]//Proceedings of the 2011 IEEE 11th International Conference on Data Mining.Vancouver:IEEE,2012:211-220.

[6] Zhu Jinghua,Yin Xuming,Wang Yake,et al.Structural holes theory-based influence maximization in social network[M]//Wireless Algorithms,Systems,and Applications.Cham:Springer,2017:860-864.

[7] Gayraud N T H,Pitoura E,Tsaparas P.Diffusion maximization in evolving social networks[C]//Proceedings of the 2015 ACM on Conference on Online Social Networks.New York:ACM,2015:125-135.

[8] Zhuang Honglei,Sun Yihan,Tang Jie,et al.Influence maximization in dynamic social networks[C]//Proceedings of the 2013 IEEE 13th International Conference on Data Mining.Shenzhen:IEEE,2014:1313-1318.

[9] Tong Guangmo,Wu Weili,Tang Shaojie,et al.Adaptive influence maximization in dynamic social networks[J].IEEE/ACM Transactions on Networking,2015,25(1):112–125.

[10] Liu Xiaodong,Liao Xiangke,Li Shanshan,et al.On the shoulders of giants:Incremental influence maximization in evolving social networks[J/OL].Complexity,2017[2017-08-01].http://dx.doi.org/10.1155/2017/5049836.

[11] Bao Yixin,Wang Xiaoke,Wang Zhi,et al.Online influence maximization in non-stationary social networks[C]//Proceedings of the 2016 IEEE/ACM 24th International Symposium on Quality of Service.Beijing:IEEE,2016:1-6.

（编辑　赵　婧）

引用格式: Zhu Jinghua,Li Yaqiong,Wang Yake,et al.Influence maximization algorithm for dynamic social networks based on linear threshold model[J].Advanced Engineering Sciences,2019,51(1):181–188.[朱敬华,李亚琼,王亚珂,等.基于线性阈值模型的动态社交网络影响最大化算法[J].工程科学与技术,2019,51(1):181–188.]