doi:10.3969/j.issn.1005-3026.2016.04.003

社交网络信息源快速定位方法

张聿博,张锡哲,徐 超,张 斌
(东北大学 计算机科学与工程学院,辽宁 沈阳 110819)

摘 要:针对在线社交网络中普遍存在的信息传播部分路径,在现有的基于观察点的信息源定位方法的基础上,提出一种基于部分路径的信息源点快速定位方法.该方法分析了利用观察点记录的部分传播路径对候选传播源点进行筛选的4种情况.通过筛选候选源点,达到了减小计算量,提高源点定位效率的目的.在模型网络上对改进算法进行实验,验证了该方法的有效性.

关 键 词:社交网络;信息传播;源定位;部分路径;候选源点筛选
中图分类号:TP 311 **文献标志码**:A **文章编号**:1005-3026(2016)04-0467-05

Fast Source Localization Method for Social Network

ZHANG Yu-bo, *ZHANG Xi-zhe*, *XU Chao*, *ZHANG Bin*
(School of Computer Science & Engineering, Northeastern University, Shenyang 110819, China. Corresponding author: ZHANG Xi-zhe, E-mail: zhangxizhe@ise.neu.edu.cn)

Abstract: Considering the phenomenon that several partial paths were recorded by the users in most social networks, a fast source localization method based on partial paths was provided by using the previous source localization method based on observers. The four cases of screening candidate sources were analyzed based on the proposed method, which made use of the partial paths recorded by the observers. By screening candidate sources, the purpose of reducing the computing expense and improving the location efficiency was achieved. The results of experiments on model network showed the effectiveness of the method.

Key words: social network; information diffusion; source localization; partial paths; candidate source screen

在线社交网络已经成为当前被广泛使用的信息传播载体之一[1],带来了巨大的商业价值和社会价值.然而,在线社交网络上的谣言传播问题也逐渐显现出来,并受到了社会的广泛关注,成为当前的研究热点之一[2-3].

如果能够及时准确地定位网络中的谣言传播源头,对控制网络上的谣言传播具有重要意义[4-5].现有的信息源定位方法主要分为两类[5]:一类是通过多次获取网络中信息传播各阶段的网络快照,利用这些信息定位信息源.Prakash 等[6]针对 SI 模型提出一种基于最小描述长度的定位方法,能够自动地确定传播源点的数目,并识别网络中的多个传播源点.Zhu 等[7]采用样本路径方法,寻找最有可能形成网络快照中样本路径的根节点作为信息源点.然而,多次获取网络传播快照需要消耗大量资源,不适用于规模庞大的社交网络.另一类方法是由 Pinto 等[8]提出的基于观察点的信息源定位方法.该方法在网络中部署少量观察点,通过观察点记录的传播信息,构建似然估计函数,计算网络中各候选源点(网络中除观察点以外的其他节点称为候选源点)的似然估计值,似然估计值最大的即为信息源.这类方法仅需要获取少量节点的传播信息,在大规模社交网络上具有较好的适用性.

收稿日期:2015-05-14
基金项目:国家科技支撑计划项目(2014BAI17B00);国家关键科技研发基金资助项目(2015BAH09F02, 2015BAH47F03);中央高校基本科研业务费专项资金资助项目(N140404011, N120804001, N120204003);国家自然科学基金资助项目(61572116, 61572117, 61502089).
作者简介:张聿博(1984-),男,辽宁沈阳人,东北大学博士研究生;张 斌(1964-),男,辽宁沈阳人,东北大学教授,博士生导师.

在实际应用中,定位效率是评价一个源点定位算法适用性好坏的重要指标. 基于观察点的信息源定位方法,其算法复杂度为 $O(N^3)$,其中 N 为网络中节点的数量. 显然,随着节点数量的增加,定位计算所需要的时间成倍增长. 对于社交网络这样的大规模网络来说,定位计算所需要的时间将大大影响其定位效率. 造成计算量过大的一个重要原因,是网络中候选源点的规模过于庞大,导致需要计算似然估计值的节点过多. 如果能够找到一种对候选源点进行筛选的方法,将有效提高定位效率.

在当前社交网络上的信息传播过程中,往往会存在这样一种现象,用户会将信息的来源附加到信息本身中,继续传播下去. 例如,在微博信息中,会用"@"符号表示这个信息在转发过程中所经过的用户. 通过这些附加信息,可以得到信息在传播过程中经过的部分传播路径,这些路径能够描述信息的真实传播过程,有效地利用这些部分路径信息可以为筛选候选源点提供依据,达到缩小候选源点范围,提高定位效率的目的.

1 传播模型与定位方法

在本文所采用的传播模型与定位方法中,G 表示整个网络,$V=\{v_1,v_2,\cdots,v_n\}$ 表示节点集,$N(v)$ 表示节点 v 的邻居节点集,θ_{uv} 表示 u 与 v 之间的传播延迟,μ 表示 G 中全部 θ_{uv} 的均值,σ 表示 θ_{uv} 的方差,$O=\{o_i\}$ 表示观察点集,t_i 表示 o_i 首次收到信息的时间,S 表示候选源点集合,$|p(u,v)|$ 表示 u 与 v 之间的最短路径的长度. 具体传播过程如下.

以 $s^*\in V$ 为源点,s^* 将消息发送给 $N(s^*)$. 对于普通节点 v,其首次收到该消息时,会将该消息发送给 $N(v)$,否则不发送. 对于观察点 o_i,其首次收到该消息时,除了转发消息,还会记录 t_i 和信息来源节点. 以此类推,直到 G 中节点全部收到该信息为止.

本文参考了 Pinto 等[8]提出的基于观察点的信息源定位方法,该方法逐个假设每个候选源点(非观察点)为实际信息源,对比各观察点收到信息的理论时间与实际时间,然后计算似然估计值 $\hat{s}$,找到似然估计值最大的候选源点即为实际信息源. 似然估计值的计算方法如下:

$$\hat{s}=\frac{\exp\left(-\frac{1}{2}(\boldsymbol{d}-\boldsymbol{\mu}_s)^{\mathrm{T}}\Lambda_s^{-1}(\boldsymbol{d}-\boldsymbol{\mu}_s)\right)}{\sqrt{|\Lambda_s|}},$$

$$[\Lambda_s]_{k,i}=\sigma^2\cdot\begin{cases}|p(o_1,o_{k+1})|, & k=i;\\ |p(o_1,o_{k+1})\cap p(o_1,o_{i+1})|, & k\neq i.\end{cases}$$

式中:$\boldsymbol{d}=[d_1,d_2,\cdots,d_k]^{\mathrm{T}}$ 是已收到信息的观察点的实际传播延迟向量;$\boldsymbol{\mu}_s$ 是理论传播延迟向量. 计算方法如下:

$$[\boldsymbol{d}]_k=t_{k+1}-t_1,$$

$$[\boldsymbol{\mu}_s]_k=\tilde{t}_{k+1}-\tilde{t}_1=\boldsymbol{\mu}\cdot(|p(s_i,o_{k+1})|-|p(s_i,o_1)|).$$

2 基于部分传播路径的信息源快速定位方法

2.1 算法思路

基于观察点的信息源定位方法[8],对候选源点构建广度优先生成树,然后计算该节点的似然估计值. 但是,如果对每个候选源点都进行这样的计算,会造成巨大的计算时间开销. 因此,如果能够在构建生成树之前,先将候选源点中真实信息源可能性很小的节点筛选掉,就可以达到减少候选源点数量,提升源点定位效率的目的.

基于观察点的信息源定位算法,其基本假设是信息沿着最短路径传播. 以此假设为基础,本文结合信息传播过程中观察点所记录的部分传播路径,提出了一种筛选候选观察点的思路. 以这些路径的首尾节点对为两端,可以得到节点对间存在的全部路径. 对于任意节点 a, b,若观察点记录了从 a 到 b 的一条路径 P,那么从 a 到 b 的其他路径长度与 P 进行比较,只能存在小于、等于、大于 3 种可能,此外,还需要考虑 P 上的节点. 因此,本文从以下 4 种情况对候选源点的筛选方法进行描述.

1) 部分传播路径上的节点. 如图 1 所示,观察点 o 记录了一条部分传播路径 1→2→3→4→5,很显然,除了节点 1 外,这条路径上的其他节点都不可能是信息源.

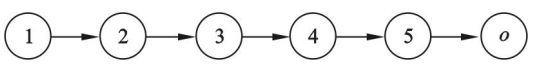

图 1 部分传播路径上的节点

Fig. 1 Nodes on partial path

2) 等于部分路径长度的路径. 如图 2 所示,观察点 o 记录了一条部分传播路径1→2→3→4→5,记为 P_1,长度为 4. 在节点 1 与节点 5 之间还存在一条路径 1→6→7→8→5,与 P_1 长度相等,记为 P_2. 如果信息源在 P_2 上,假设是节点 6,根据最短路径假设,信息从节点 6 传播到节点 5 的路径应该是 6→7→8→5,该路径的长度小于 P_1. 若节

点 6 是信息源，那么节点 5 应该首先接收到节点 8 传入的信息，这与观察点记录的传播信息不符. 同理，P_2 上的其他节点也是这样的结果. 可以得出，网络中与记录的部分传播路径有着相同首尾节点的等长路径上的节点（除首节点），是信息源的可能性很小.

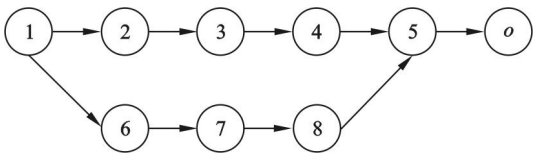

图 2　等于部分路径长度的路径
Fig. 2　Paths equal to partial path

3）小于部分路径长度的路径. 如图 3 所示，观察点 o 记录了一条部分传播路径 1→2→3→4→5，记为 P_1，长度为 4. 在节点 1 与节点 5 之间还存在一条路径 1→6→7→5，小于 P_1 的长度，记为 P_2. 与情况 2 类似，如果信息源在 P_2 上，那么节点 5 应该首先接收到节点 7 传入的信息，这与观察点记录的传播信息不符. 可以得出，网络中与记录的部分传播路径有着相同首尾节点，但长度小于该路径的路径上的节点（除首节点），是信息源的可能性很小.

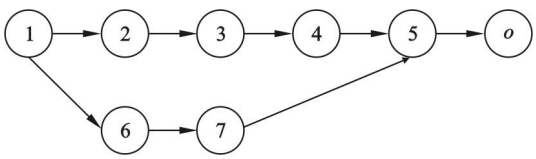

图 3　小于部分路径长度的路径
Fig. 3　Paths shorter than partial path

4）大于部分路径长度的路径. 如图 4 所示，观察点 o 记录了一条部分传播路径 1→2→3→4，记为 P_1，长度为 3. 在节点 1 与节点 4 之间还存在一条路径 1→5→6→7→8→9→4，大于 P_1 的长度，记为 P_2. 通过对情况 2 与情况 3 的分析，网络中节点 6，7，8，9 不可能是信息源点. 图 4 中只给出了以节点 1 与节点 4 为首尾的路径，若以节点 5 为首节点时，存在路径 5→1，那么就存在一条路径 5→1→2→3→4，该路径小于5→6→7→ 8→9→4 的长度，那么节点 5 就可能是信息源. 可以得出，大于部分路径长度的路径为 P，长度为 $|P|$，部分路径为 P'，长度为 $|P'|$，则筛选出的候选源点是沿着 P 路径方向第 $|P| - |P'| - 1$ 节点到第 $|P| + 1$ 节点之间的所有节点.

2.2　算法描述

根据上节所述的筛选方法，本文对基于观察点的信息源定位方法进行改进. 与原算法相比，本文所提出的定位方法，结合观察点记录的部分传播路径，先对候选源点进行筛选，缩小候选源点的范围，然后再进行定位计算. 其中，筛选候选源点的步骤如下.

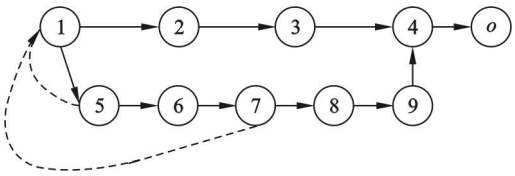

图 4　大于部分路径长度的路径
Fig. 4　Paths longer than partial path

步骤 1　设观察点集合为 O，候选源点集合为 C. 遍历 O，获取所有部分路径，并加入集合 P.

步骤 2　对 P 中的每个部分路径 P_i，获取网络所有与其首尾节点相同的路径，并加入集合 T.

步骤 3　遍历 T，取出 T 中的一条路径 T_i，如果路径长度 T_i 小于等于部分路径长度 P_i，则将 T_i 中首节点以外的所有节点加入集合 F. 如果 $|T_i| > |P_i|$，则将 T_i 中按 T_i 的方向顺序第 $|T_i| - |P_i| - 1$ 节点到第 $|T_i| + 1$ 个节点加入集合 F.

步骤 4　集合 C 与 F 做差集得到集合 O_f，即为筛选后的候选源点集合.

具体伪代码如下：

```
observers = graph. getObervers( ) //获取所有观察点
candidate = graph. getNodes( ) - observers;
for(0≤i≤observers. size) do
Node n = observers. get(i);
fragments = n. getFragments( );//获取所有部分路径
for(0≤f≤fragments. size) do
nodesShorter = fragments. getShortestPath(i);
nodesEqual = fragments. getShortestPath(i);
nodesLonger = fragments. getShortestPath(i);//
将全部符合条件的路径上的节点加入过滤集合
filter. add(nodes in shorter paths);
    filter. add(nodes in equal paths);
    filter. add(nodes in longer paths);//
部分路径上除首节点以外节点加入过滤集合
    filter. add(nodes in partial paths);
    end for
  end for
filteredCandidate = candidates - nodes in filter ;//得到筛选后的候选源点集合
return;
```

3 实验与分析

为了验证快速定位算法的定位效率及影响因素,本文选用经典的 ER 模型[9]网络和 BA 模型[10]网络进行实验. 具体实验数据如表 1 所示,其中 N 表示网络节点数,L 表示边数,$<k>$ 表示网络平均度,d 表示网络直径,Apl 表示网络平均路径长度.

在实际应用中,不是所有用户会记录传播路径,并且所记录的传播路径长度也不同. 本文设能够记录传播路径的观察点占观察点总数的比例为 f_p,部分路径的最大长度为 F,观察点的选取策略按照网络中节点的入度排列,选取入度大的节点作为观察点,观察点占全部节点的比例为 p. 在下面的各部分实验中,p 的取值范围为 0.05 ~ 0.4(递加 0.05),f_p 的取值范围为 0.05 ~ 0.3(递加 0.05),F 取值为 3.

表 1 实验数据

Table 1 Experimental data

网络	N	L	$\langle k\rangle$	d	Apl
ERNetwork1	6 000	180 804	30.134	14	3.092
ERNetwork2	8 000	256 690	32.086	17	3.160
BANetwork1	6 000	150 000	25	5	3.010
BANetwork2	8 000	200 000	25	5	3.098

1)改进算法与原算法定位准确率比较. 为了考查改进算法对原算法(基于观察点的信息源定位方法)定位准确率的影响,选取 ERNetwork1 与 BANetwork1 作为实验数据集. 对每种观察点比例下取每种 f_p 做 1 000 次实验,得到对应的定位命中率,实验结果如图 5 所示.

由图 5 可知,在各观察点比例下,改进的快速定位算法与原算法的定位准确率相差很小,可以忽略不计. 此外,改进算法的命中率随着观察点比例的增大而升高,趋势也与原算法的定位准确率变化趋势相吻合.

2)改进算法的候选源点筛选效果. 为了分析改进算法对定位效率的提升,选取 ERNetwork1-2,BANetwork1-2 作为实验数据集. 对每种观察点比例下取每种 f_p 做 2 000 次实验,得到对应的筛选比均值(筛选出的节点占原来候选节点的比例),实验结果如图 6 所示. 分析实验结果可知,部分传播路径比例与筛选比成正比,对于同一观察点比例,f_p 越大筛选比越高. 这是因为 f_p 越大,观察点获取的部分传播路径信息越多,可以筛选掉的候选源点越多.

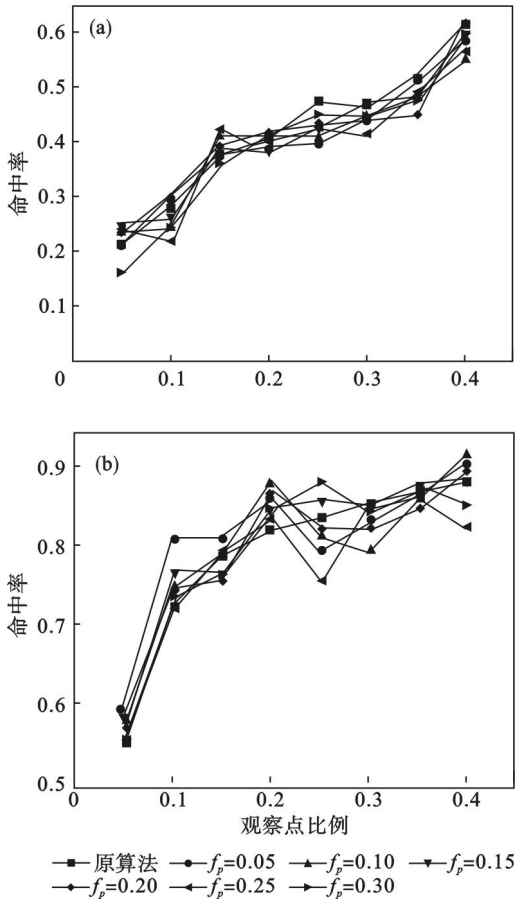

图 5 改进算法与原算法定位准确率比较

Fig. 5 Comparison of localization accuracy between improved and original algorithms

(a)—ERNetwork1; (b)—BANetwork1.

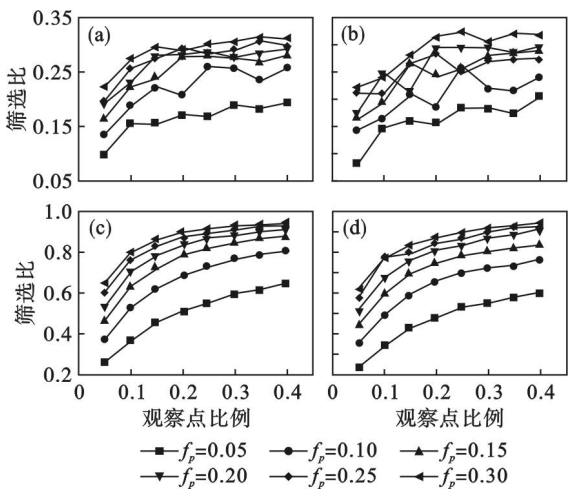

图 6 改进算法的候选源点筛选结果

Fig. 6 Results of screening candidate source nodes with improved algorithm

(a)—ERNetwork1; (b)—ERNetwork2; (c)—BANetwork1; (d)—BANetwork2.

3)改进算法与原算法定位时间比较. 改进算法最主要的作用就是提升原算法的定位效率,为

了考察改进算法的定位效率提升效果，选择 ERNetwork1 与 BANetwork1 作为实验数据. 对每种观察点比例下取每种 f_p 做 2 000 次实验，得到对应的定位效率提升比例（原算法与改进算法的平均定位时间之间差除以原算法的平均定位时间），实验结果如图 7 所示. 可以得出，在 ER 模型网络上，改进算法对原算法的定位效率有 5% ~ 25% 左右的提升；在 BA 模型网络上，改进算法对原算法的定位效率有 15% ~35% 左右的提升. 并且提升效率随观察点比例的增大而增大. 观察点比例在 0. 05 ~0. 2 左右时，效率上升趋势明显，当观察点比例达到 0. 25 后，效率增速减缓，并在某一值范围内波动，且 f_p 值越大，其定位效率提升越大.

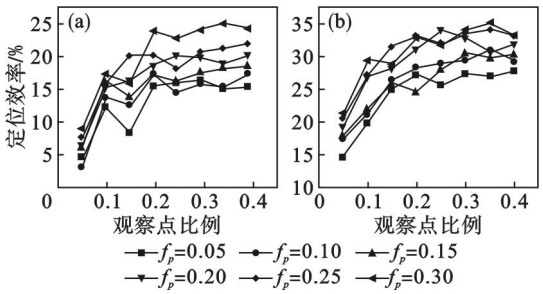

图 7　改进算法的定位效率提升结果

Fig. 7　Results of localization efficiency improvement with improved algorithm

(a)—ERNetwork1；(b)—BANetwork1.

4　结　　论

本文针对现有的基于观察点的信息源定位方法定位效率较低，不适用于大规模社交网络这一问题，结合社交网络信息传播过程中会记录部分传播路径这一现象，提出了一种基于部分传播路径的信息源快速定位方法. 该方法对传播模型及基于观察点的信息源定位方法进行分析，得出了通过已知部分路径对候选源点进行筛选的方法，以达到减少定位计算量，提高定位效率的目的. 在模型网络上进行实验，结果表明，本文所提出的改进算法能够在基本不影响定位命中率的情况下，有效提高定位效率.

参考文献：

[1] Dong W, Zhang W, Tan C W. Rooting out the rumor culprit from suspects [C]// IEEE International Symposium on Information Theory. Istanbul: IEEE, 2013: 2671 - 2675.

[2] Budak C, Agrawal D, El-Abbadi A. Limiting the spread of misinformation in social networks [C]//Proceedings of the 20th International Conference on World Wide Web. Hyderabad: ACM, 2011: 665 - 674.

[3] Shah D, Zaman T. Detecting sources of computer viruses in networks: theory and experiment [C]//ACM SIGMETRICS Performance Evaluation Review. New York: ACM, 2010: 203 - 214.

[4] Lokhov A Y, Mezard M, Ohta H. Inferring the origin of an epidemic with dynamic message-passing algorithm [J]. *Physical Review E*, 2014, 90(1): 012801.

[5] 张聿博，张锡哲，张斌. 面向社交网络信息源定位的观察点部署方法[J]. 软件学报，2014，25(12)：2837 - 2851.
(Zhang Yu-bo, Zhang Xi-zhe, Zhang Bin. Observer deployment method for locating the information source in social network [J]. *Journal of Software*, 2014, 25(12): 2837 - 2851.)

[6] Prakash B A, Vreeken J, Faloutsos C. Spotting culprits in epidemics: how many and which ones? [C]// IEEE 12th International Conference on Data Mining. Brussels: IEEE, 2012: 11 - 20.

[7] Zhu K, Ying L. Information source detection in the SIR model: a sample path based approach [J]. *IEEE/ACM Transactions on Networking*, 2014, 11(20): 1 - 14.

[8] Pinto P C, Thiran P, Vetterli M. Locating the source of diffusion in large-scale networks [J]. *Physical Review Letters*, 2012, 109(6): 068702.

[9] Erd6s P, Rényi A. On random graphs I [J]. *Publications Mathematician*, 1959, 6: 290 - 297.

[10] Barabási A L, Albert R. Emergence of scaling in random networks [J]. *Science*, 1999, 286(5439): 509 - 512.