基于信息熵与迭代因子的复杂网络节点重要性评价方法

汪亭亭 梁宗文† 张若曦

(西南石油大学计算机科学学院, 成都 610500)

(2022 年 9 月 27 日收到; 2022 年 11 月 27 日收到修改稿)

在复杂网络的研究中, 如何有效地衡量节点的重要性一直都是学者们关心的问题. 在节点重要性研究领域, 基于拓扑学信息来判断节点重要性的方法被大量提出, 如 K-shell 方法. K-shell 是一种寻找可能具有重要影响力节点的有效方法, 在大量的研究工作中被广泛引用. 但是, K-shell 过多地强调了中心节点的影响力, 却忽视了处于网络外围节点作用力的影响. 为了更好地衡量网络中各个节点对传播的促进作用, 本文提出了一种基于迭代因子和节点信息熵的改进方法来评估各个层次节点的传播能力. 为评价本文方法的性能, 本文采用 SIR 模型进行仿真实验来对各节点的传播效率进行评估, 并在实验中将本文算法和其他算法进行了对比. 实验结果表明, 本文所提方法具有更好的性能, 并且适合解决大规模复杂网络中的节点重要性评价问题.

关键词: 重要节点, 迭代因子, 信息熵, 复杂网络

PACS: 89.20.Ff, 02.10.Ox, 89.75.Fb **DOI:** 10.7498/aps.72.20221878

1 引 言

在网络科学领域中, 研究节点重要性的排序算法一直都是学者们追随的热点话题, 其目的是为了通过对节点的重要性排序寻找出对传播起关键性作用的节点. 在病毒网络中通过对重要节点的及时控制可以抑制病毒大面积的扩散[1], 在社交网络中, 商家可以把新产品投放到重要客户中, 通过重要客户的宣传实现投资效益最大化[2]. 由此可以看出研究网络中的重要节点不仅有重要的理论意义, 更有重要的现实意义.

目前已经提出了各种方法来对节点的重要性进行排序. 在网络结构拓扑基础上发展起来的经典算法有度中心性 (DC)[3]、介数中心性 (BC)[4]、接近中心性[5] 和 h 指数[6] 等, 这些都是基于网络拓扑结构的经典算法[7]. 另外, pagerank[8] 和 leaderank[9] 是基于随机游走的两个代表性方法. Kitsak 等[10] 提出了 K-shell 方法, 该方法的算法实现非常简单, 有研究显示该算法对识别影响力节点具有显著的度量作用[11]. 但是 K-shell(ks) 索引对网络拓扑全局信息要求较高且在单调性 (排名列表中拥有相同排名节点的比例) 上表现不佳[12], 即在同一个 K-shell(ks) 值中的所有节点都拥有相同的排名, 这样不利于唯一地区分节点的排名. 之后研究者们在 K-shell 基础上提出了很多改进的方法. Basaras 等[13] 提出了基于混合度和 K-shell 的算法, 该方法提出了 μ幂社区指数 (μ-power community index, μ-PCI). 它是 K-shell 和中心度的混合, 该算法以完全局部化的计算方式达到了适用于任何类型的网络. Wang 等[14] 利用 k-shell 的迭代信息来区分具有相同 ks 值的节点, 并同时考虑了节点度来综合量化节点的重要性, 具有很好的准确性. 网络中少量的节点具有大量的边, 这些节点也被称为“富节

† 通信作者. E-mail: zongwen-liang@hotmail.com

点”, 它们会出现倾向于彼此之间相互连接的现象, 这种现象一般被称为富人俱乐部现象[15]. 如果通过排序方法选出来的重要节点作为种子节点, 种子节点之间具有高度连接, 就会受富人俱乐部现象的影响, 造成大量的活跃节点在传播时出现交叉现象, 传播仅在小范围内扩散. 而这些以 K-shell 为基础的排序方法, 往往无法避免网络中富人俱乐部现象带来的影响.

有研究学者已经提出很多方法来规避富人俱乐部现象的带来的影响, 使得基于拓扑排序的方法变得更可靠. 针对富人俱乐部现象问题, Wang 等[16]提出一种改进的 K-shell 方法 (improved K-shell method, IKS), 该方法通过迭代筛选出 K-shell 各层中信息熵最高的节点, 从而有效地避免富人俱乐部现象, 实验表明其对网络中前 K 个节点的传播影响力衡量更准确. 但在同一 shell 内有大量信息熵相等的节点时, 该算法会随机选取其中之一并把其余节点投入到下次迭代当中, 这就造成了本来排名靠前的节点因无限迭代而靠后. 在 Zareie 等[17]所提出的算法中考虑了节点及其邻域集的公共层次, 将迭代因子 (iteration, IT) 应用于网络分层中, 使得网络中节点更具有差异性, 从而提出一种基于邻域相关系数的关键节点识别算法.

受 Zareie 等[17]所提算法的启发, 本文沿用迭代因子 (iteration, IT) 对网络进行分层; 在改进的 K-shell 方法 (improved K-shell method, IKS)[18]的基础之上, 利用迭代因子来对网络的结构进行分层, 然后再分别计算每层中节点的信息熵, 提出了基于迭代因子和信息熵相结合的方法 (简称 IE_{+}) 来衡量网络中的节点重要性, 该方法对在迭代过程中因随机选择造成节点排序靠后的问题有所改进, 同时在具有富人俱乐部现象的网络中进行节点重要性排序时也具有较好的表现. 本文在八种常见网络数据中使用 SIR 模型[18,19]来模拟病毒传播的过程, 将所提出的算法与常见算法进行比较, 实验结果表明, 本文所提方法具有更好的性能, 并且适合解决大规模复杂网络中的节点重要性评价问题.

本文的其余部分安排如下. 第 2 节简要叙述了现有的一些经典算法. 第 3 节中将详细阐述本文的算法思想. 数据集将在第 4 节中介绍. 第 5 节将简要介绍评价指标. 实验设置、结果和讨论在第 6 节中提及. 最后, 在第 7 节中给出结论.

2 相关工作

一个无向未加权网络通常表示为 $G=(V, E)$, 其中 V 和 E 分别表示节点和边的集合. 它也可以定义为一个邻接矩阵 $\boldsymbol{A}=(a_{ij})_{n\times n}$, 如果节点 v_i 和 v_j 有一条边相连接, 则 $a_{ij}=1$, 否则 $a_{ij}=0$.

大部分算法都是基于拓扑结构, 关注节点的中心性. 此前学者们提出了许多中心性度量方法, 这些方法从不同的角度衡量了节点的重要性. 在这里, 简要回顾几个中心性指标的定义.

在度中心性算法中, DC 算法[3]主要考虑了节点度中心性, 并得出节点邻居数量越大传播能力越强; 接近中心性 (CC)[5]算法则更关注节点和整个网络之间的关系, 认为节点与网络中所有节点之间距离的平均值越小, 节点越重要; 学者们还提出了相对新颖的方法, 例如基于重力的方法论上, 取两个节点的 ks 值作为质量, 两个节点之间的最短路径作为距离[20,21], 两个节点之间的相互作用关系随着他们的距离而减小, 模仿重力公式将两个节点间的 ks 值的乘积与两节点间的最短距离的比值作为衡量节点传播能力的度量, 从而实现对节点重要性的排序.

Kitsak 等[10]提出了 K-shell 方法, 该方法认为节点的影响力是由它的位置决定的, 而最有影响力的节点应该是网络的核心. K-shell 分解是一个迭代过程, 第一步是删除所有度数为 1 的节点, 直到网络中没有度数为 1 的节点, 被移除节点 ks = 1. 第二步是从网络中移除所有度数为 2 的节点, 直到网络中没有度数为 2 的节点, 被移除节点 ks = 2. 迭代继续, 直到所有节点都从网络中删除. 图 1 列出了一个包含 26 个节点和 32 条边的网络图. 通过 K-shell分解得到每个节点的 ks 值. K-shell 算法认为 ks值越大, 传播影响越大. 这意味着在图 1 所示网络中, 节点 1, 2, 3, 4, 5 的影响力最大, 而 ks = 1 的节点传播影响力最小. 在 K-shell 的分解过程中, 对度很大但位于网络边缘节点的影响力衡量不够准确, 例如图 1 中的节点 21. 研究人员基于 K-shell 提出了大量的扩展方法, 如邻域核心中心性 (C_{nc})、扩展邻域核心中心性方法 ($C_{\mathrm{nc}+}$)[22]等算法认为, 一个节点的影响不仅取决于它的自己的 ks 值, 也依赖于其邻居节点的 ks 值.

最近, 一些混合的度量方法被陆续提出. 这些方法充分利用节点的拓扑信息, 利用混合的衡量

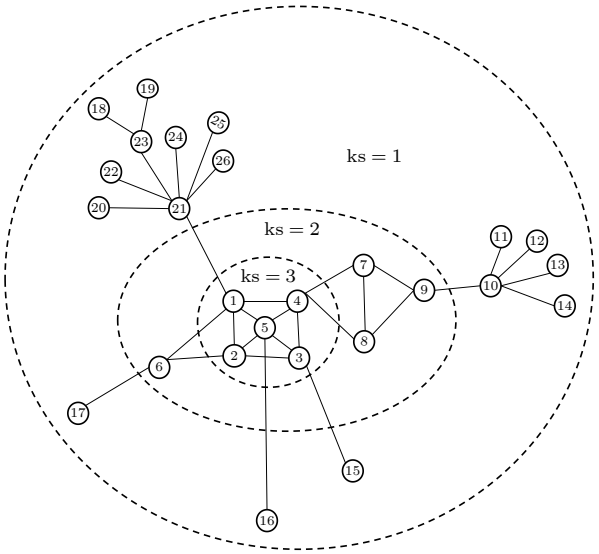

图 1 一个简单示例图

Fig. 1. A sample graph.

指标从不同的角度来对节点的重要性进行评价. Bha 等 [23] 提出提高的混合排序方法, 该方法结合了两个中心性——扩展的邻居中心性 ($C_{\mathrm{nc}+}$) 和 h 指数中心性, 该方法在排序准确性上具有很好的性能. 阮逸润等 [24] 提出了基于结构洞引力模型的改进算法, 综合考虑节点 h 指数、节点核数以及节点的结构洞位置.

除此之外, 在动态网络中, 网络拓扑随着时间的推移而不断变化, 导致动态网络中关键节点的识别变成一项艰巨的任务. 研究者们通过不同时间段的网络快照信息对节点进行排名 [25−30], Qu 等 [31] 提出时态网络中的时态信息收集 (TIG) 过程, 将 TIG 过程作为节点的重要性度量, 可用于节点重要性排名. 胡钢等 [32] 依托复杂网络的层间时序关联耦合关系、层内连接关系和层间逼近关系构建时序网络超邻接矩阵, 提出了基于时序网络层间同构率动态演化的超邻接矩阵建模的重要节点辨识方法.

无论是静态网络还是动态网络, 它们都遵循幂律网络中的一个重要性质就是少数几个节点连接数量较多, 而连接数量较多的节点更愿意去连接比它连接数量更多或同等多的节点, 这时就造成了富人俱乐部现象的出现. 为了避免富人俱乐部现象对排序结果带来的影响, Liu 等 [27] 提出了一种基于每个节点的局部索引秩 (local index rank, LIR) 的算法, 可以对具有富人俱乐部现象的网络中的节点实现快速排序; Namtirtha 等 [28] 提出了一种混合指标的度量方法, 考虑了节点的度、节点间的联合影响率和节点间的最短距离, 从网络中心到网络边缘来对节点进行排序; 改进的 K-shell 算法 [16] 采用 K-shell 分层与节点信息熵相结合的方法来迭代地选取重要节点, 考虑了不同 shell 层中网络节点的重要性, 但该算法会在具有相同信息熵的节点中随机选择一个并把其余节点投入到下次迭代当中, 这就造成了本来排名靠前的节点因无限迭代而靠后. 受 IKS 算法的启发, 本文提出了一种基于迭代因子和信息熵的算法来识别从网络外围到核心的每个网络级别的重要节点, 将此方法简称为 IE_{+}.

3 算法介绍

3.1 迭代因子

在 IKS 方法中, 网络使用 K-shell 进行分层. 从图 1 可以看出, 虽然节点 7, 8 和 9 在同一个 K-shell 中, 但是节点 7, 8 较节点 9 更接近网络的核心, 当网络中的节点被移除时, 节点 9 在节点 7, 8 之前被移除. 与被感染的节点 9 相比, 被感染的节点 7, 8 可以达到更好的传播效果. 鉴于 K-shell 对节点重要性度量不够完善, 本文提出迭代因子 (iteration, IT) 对节点进行分层, 使得网络层次结构更加清晰.

以图 1 为例, 设置迭代因子的初始值 IT=1, 然后将度数最小的节点从网络中移除. 这些被移除的节点属于图结构的第一层, 然后在本次迭代中被删除节点的迭代因子 IT = 1. 在下一阶段, IT 值增加 1, 并再次从图中删除度最小节点, 并将在本次迭代中被删除节点的 IT = 2. 这个过程会一直迭代, 直到图中没有剩余节点. 在每次迭代中, IT 的值都会加 1 并分配给在此阶段中被删除的节点. 从图 1可以看出, 节点 7, 8, 9 不再处于同一结构层级, 但节点 7 和 8 更靠近网络中心, 因此比节点 9 具有较大的 IT 值. 同理, 对位于网络中心的节点 (如节点 1, 2, 3, 4, 5) 的网络层次划分更细, 节点 5 被认为是网络的核心, 节点 1, 2, 3 和 4 被认为是次要核心.

3.2 信息熵

在 IKS 算法中, 信息熵被定义为

$$e_i = -\sum\nolimits_{j\in\Gamma(i)} I_j \cdot \ln I_j, \tag{1}$$

其中 $j \in \Gamma(i)$ 是节点 v_i 的邻居节点的集合, 其中:

$$I_i = \frac{k_i}{\sum_{j=1}^{N} k_j}. \tag{2}$$

k_i是节点 vi 的度, $k_i = \sum_{j=1}^{N} a_{ij}$.

从 (1) 式和 (2) 式可以看出, 节点信息熵的计算主要依赖于节点的本地邻居信息. 节点 17, 18 和 19 都在网络的外围, 如果仅依靠邻居的度信息来计算节点的信息熵, 那么这三个节点的信息熵和 ks 值均相同. 但节点 17 的邻居节点比其他两个节点的邻居节点更接近网络的中心, 因而仅依靠邻居节点的度信息显然是不够的. 因此, 本文结合节点在网络中的位置, 基于 ks 提出一种计算节点信息熵的新方法:

$$e_{i+} = -\sum_{j\in\Gamma(i)} I_j \cdot \ln I_j \cdot \mathrm{ks}_j, \tag{3}$$

其中 (i) 是节点 v_i 的邻居集合; I_j 的定义如 (2) 式所示; $\mathrm{ks_j}$ 表示节点 j 的 ks 值. 以节点 17 为例计算其信息熵:

$$\begin{aligned} e_{17+} &= -\sum_{j\in\Gamma(i)} I_j \cdot \ln I_j \cdot \mathrm{ks}_j \\ &= -\sum \left(\frac{3}{64} \times \ln \frac{3}{64} \times 2 \right) \\ &= 0.2219. \end{aligned} \tag{4}$$

节点 v_{17} 的度数为 1, 其邻居节点为 v_6, 该节点的 ks 值为 2, 其度数为 3, 经计算节点 17 的节点熵为 0.2219. 同理, 可以计算网络中其他节点的信息熵, 计算结果如表 1 和表 2 所示. 在改进的信息熵计算方法中, 由于一个节点的信息熵不仅与它的邻居度信息有关, 还与它在网络中的位置相关. 通过对比表 1 和表 2, 可以发现改进的信息熵计算方法能更清晰地区分节点的重要性程度.

3.3 算法步骤

本文基于迭代因子 (IT), 通过 ks 值对节点信息熵的计算进行改进, 提出了迭代因子和信息熵相结合的算法 (简称 $\mathrm{IE_+}$) 来对节点的重要性程度进行度量, 该算法的步骤如下:

1) 使用 K-shell 算法计算网络中所有节点的 ks 值, 然后令 IT = 1;

2) 将当前网络中度最小的节点的迭代因子记为 IT, 并根据 (4) 式来计算这些节点的信息熵 $\mathrm{e_{i+}}$;

3) 从网络中删除这些度最小的节点;

4) 若网络中的节点个数为 0, 记录 IT(max)= 前迭代因子 IT, 跳转步骤 5; 否则, IT 加 1, 跳转步骤 2;

5) 选择当前迭代因子 IT 对应节点集合中信息熵最大的节点, 按序放入节点重要性排序集合中. 如果有多个节点信息熵值相等时, 按照节点的序号从大到小将所有信息熵相等的节点全部放入重要性排序集合中;

表 1 节点在每个 shell 中的信息熵
Table 1. Information entropy of each node.

ks	Node	E
3	1	0.9571
	4	0.8565
	5	0.8099
	2	0.7151
	3	0.6366
2	7/8/9	0.4861
	6	0.4374
1	21	0.6675
	10	0.4034
	23	0.3720
	20/22/24/25/26	0.2420
	11/12/13/14/16	0.1992
	15	0.1733
	17/18/19	0.1435

表 2 节点在每个迭代层中的信息熵
Table 2. Information entropy of each node.

Iteration	Node	E_+
7	5	1.3728
6	1	1.4579
	4	1.4378
	2	1.2159
	3	1.0589
5	7/8	0.7839
4	6	0.7189
	9	0.6430
3	21	0.8084
2	10	0.4818
	23	0.3720
1	16	0.3400
	15	0.3139
	20/22/24/25/26	0.2420
	17	0.2219
	11/12/13/14	0.1992
	18/19	0.1435

6) 如果 IT = 0, 则跳转步骤 7, 否则 IT 减 1, 跳转步骤 5;

7) 如果网络中所有节点都已经被放入重要性节点排序集合中, 则结束算法; 否则, 令前迭代因子 IT = IT(max), 跳转到步骤 5.

算法伪代码如下所示:

IE$_+$算法伪代码

输入: 网络结构$G=(V, E)$

输出: 网络中节点的排序索引Rank

通过$G=(V, E)$得出邻接矩阵$\boldsymbol{A}$

通过K-shell算法得出每个节点的ks值

IT $\leftarrow$ 1

while $|V|$ do

$V_{\text{temp}} \leftarrow \{\ \}$

$V_i.k \leftarrow \sum_{j=1}^{N} a_{ij}$

mindegree $\leftarrow$ min($V.k$)

$V_{\text{temp}} \leftarrow$ find ($V.\ k ==$ mindegree)

while V_{temp} do

V_{temp}. IT$\leftarrow$IT

$V_{\text{temp}}.e_+ \leftarrow -\sum_{j\in\Gamma(i)} I_j \cdot \ln I_j \cdot \text{ks}_j$

endwhile

delete(V_{temp})

IT$\leftarrow$IT+1

$V \leftarrow V - V_{\text{temp}}$

endwhile

ITMax $\leftarrow$ IT

while length(Rank) $<$ N do

for IT$\leftarrow$ITMax to 1 do

V_{temp} = find(max(V.IT.e_+))

if length (V_{temp}) $>$ 1

按节点序号从大到小排序

end if

Rank $\leftarrow$ { V_{temp}, Rank}

end for

end while

return Rank

计算出每个节点的 ks 值和迭代因子 IT 后, 将迭代因子相同的节点按照信息熵值降序排列, 如表 2 所列. 然后对节点进行排序, 在最大迭代因子中选择最大信息熵的节点, 显而易见应该选择节点 5, 然后在下一个迭代因子中选择节点 1. 在下一层中节点 7 和 8 具有相同的改进的信息熵值. 按节点序号从大到小顺序放入, 直到最小的迭代因子层次结构中选择信息熵最大的节点. 此时, 第一次迭代结束, 下一次迭代继续从迭代因子最高中选择节点, 直到所有节点都被选中.

在表 3 中, 使用了不同的方法对图 1 中的网络进行排序. 因为每种方法对节点重要性的识别原理不同, 可以看出每种方法的排序结果略有不同. 与 IKS 算法相比, 在迭代次数相同的情况下, 本文算法识别出的重要节点在网络中的分布更广, 这表明本文算法能更有效地避免在迭代次数相同时出现的富人俱乐部现象.

4 数据集

选择了八种不同类型的网络, 其详细信息见表 4. 1) NS 是一个由从事网络科学工作的科学家组成的合作网络[33]. 2) EEC 描述了一家大型欧洲研究机构成员之间的电子邮件交换[34]. 3) PB 是美国政治博客的网络[35]. 4) Facebook 描述了该网站的社交圈[36]. 5) WV 是一个维基百科网络, 描述了投票记录[37]. 6) Sport 是从体育网络收集的有关 Facebook 页面上的体育运动的信息 (2017 年 1 月)[38]. 7) Sex 是一个二分网络, 其中节点是女性和男性. 当男性写帖子表明与女性发生性接触时, 他们之间的联系就会建立起来[39]. 8) CondMat 是一个协作网络, 涵盖了凝聚态类别中作者论文之间的科学合作关系[40].

5 评价指标

5.1 SIR

在本文中, 使用 SIR 模型[18,19] 来验证算法的表现能力. 通过模拟 SIR 模型的传播过程, 可以得到每个节点的传播能力. 在 SIR 模型中, 每个节点可以具有三种状态, 即易感状态、感染状态和恢复状态. 一开始, 网络中的所有节点都处于易感状态, 除了原始的受感染节点. 在每个时间段中, 每个被感染的节点都会以 β 的概率感染那些处于易感状态的邻居节点. 同时, 受感染节点将以 λ 的概率进入恢复状态并不会再次感染, 当网络中没有受感染节点时, 此传播过程结束. 在选择传播值 β 时, 它可以略大于网络流行阈值 $\beta_{\text{th}}=\langle k\rangle/\langle k^2\rangle$, 其中 k 是平均度, k^2 是二阶平均度[41]. 不同网络中的 β_{th} 和 β_{c} 值如表 4 所列. 当网络达到稳定状态时, 记录恢复

的节点总数, 可以用来衡量节点的传播能力, 对每个节点重复该过程来衡量它的传播效率. 为了获得更准确的实验数据, SIR 模型传播过程的模拟次数由网络规模决定, 在 $N<10^4$ 的小型网络中模拟次数为 1000 次, 在 $N \geqslant 10^4$ 大型网络中模拟次数为 100 次.

5.2 相关系数

为了验证本文算法的性能, 使用 Kendall Tau[42] 系数 τ 来衡量不同算法得到的节点重要性排名表与 SIR 模型模拟的排名表之间的相关性, τ 定义为

$$\tau(R)=\frac{2(N_{\rm c}-N_{\rm d})}{N(N-1)}, \tag{5}$$

其中 $N_{\rm c}$ 和 $N_{\rm d}$ 分别是经过计算后相关性一致和不一致的数量. 考虑具有 N 个节点的两个相关序列 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$, $\boldsymbol{X}=(x_1, x_2, \cdots, x_n)$ 和 $\boldsymbol{Y}=(y_1, y_2, \cdots, y_n)$. 任何一对二元组 (x_i, y_i) 和 $(x_j, y_j)(x \neq y)$, 当 $x_i>x_j$和 $y_i>y_j$ 或 $x_i<x_j$ 和 $y_i<y_j$ 这两个元素被认为是一致的, 如果 $x_i>x_j$ 和 $y_i<y_j$ 或 $x_i<x_j$ 和 $y_i>y_j$ 它们是不一致的, 如果 $x_i=x_j$ 或 $y_i=y_j$ 时不计入 $N_{\rm c}$ 和 $N_{\rm d}$. 系数必须在–1 $\leqslant \tau \leqslant$ 1 的范围内, τ 值越大, 算法的排序结果越接近准确值.

表 3 由不同方法得出的排名: DC, CC, ks, $C_{\rm nc}$, $C_{\rm nc+}$, IKS, $\rm IE_+$

Table 3. The ranking lists determined by different methonds: DC, CC, ks, $\rm C_{nc}$, $\rm C_{nc+}$, IKS, $\rm IE_+$.

Rank	DC	CC	ks	$C_{\rm nc}$	$C_{\rm nc+}$	IKS	$\rm IE_+$
1	21	1	1—5	4, 5	1	1	5
2	1, 4, 5, 10	4	6—9	1	4, 5	7	1
3	2, 3	5	10—26	2	2	21	8
4	6—9, 23	21		3	3	4	7
5	11—20	7, 8		21	6—8	9	6
6	22, 24—26	6		6—8	9, 21	10	21
7		9		10	16	5	10
8		23		9	23	8	16
9		16, 20, 22, 24—26		15, 16, 23	15	23	4
10				17	10, 20, 22, 24—26	2	9
11				others		6	23
12						26	15
13						3	2
14						20, 22, 24, 25	20, 22, 24—262
15						11—13, 14, 16	3
16						15	17
17						17—19	11—14
18							18, 19

表 4 八个常见网络的基本拓扑特征, N 和$|E|$是节点和边的数量, $\langle d\rangle$和 $\langle k\rangle$ 是平均距离和平均度, c 是聚类系数, $\beta_{\rm th}$ 和 $\beta_{\rm c}$ 是流行阈值和传播值

Table 4. The basic topological features of the eight real neworks, N and $|E|$, $|E|$ are the number of nodes and edges, $\langle d\rangle$ and $\langle k\rangle$ are the average distance and the average degree, c is the clustering coefficient, $\beta_{\rm th}$ and $\beta_{\rm c}$ are the epidemic threshold and the spread value.

Network	N	$\lvert E\rvert$	$\langle d\rangle$	c	$\langle k\rangle$	$\beta_{\rm th}$	$\beta_{\rm c}$
NS	379	914	6.0419	0.7981	4.8232	0.1247	0.2494
EEC	986	16064	2.5869	0.4505	32.5842	0.0134	0.0268
PB	1222	16714	2.7375	0.3600	27.3552	0.0123	0.0246
Fecebook	4039	88234	3.6925	0.6170	43.6910	0.0094	0.0188
WV	7066	100736	3.2475	0.2090	28.5129	0.0069	0.0138
Sport	13866	86858	4.2748	0.2761	12.5281	0.0260	0.0520
Sex	15810	38540	7.4630	0	4.8754	0.0365	0.0730
CondMat	23122	93497	5.3523	0.6334	8.0835	0.0450	0.0900

5.3 单调关系

一个好的节点重要性排序算法应该是每个节点都被分配唯一的排名索引, 如果在同一排名索引上出现多个节点, 那么这样的算法被认为是存在缺陷的. 为了定量测量不同指标的分辨率, 使用了排名列表的单调性指标 $M(R)$[22]:

$$M(R)=\left[1-\frac{\sum_{r\in R}N_r(N_r-1)}{N(N-1)}\right]^2, \tag{6}$$

其中 N_r 是具有相同索引值 r 的节点数. 如果 $M(R)=1$, 表示该算法是完全单调的, 并且每个节点被归类为不同的索引值, 如果 $M(R)=0$, 则所有节点处于同一等级. 单调性指标反映排序算法是否能很好地将节点区分开来.

5.4 平均最短路径长度

计算每对传播者之间的平均最短路径长度[43], 这是一个基本指标. 如果每个节点感染其他节点的概率相同, 则初始感染节点越分散, 传播范围越广. 本文选择初始节点 S 作为度量, 其定义为

$$L_s=\frac{1}{|s|(|s|-1)}\sum_{\substack{u,v\in s\\ u\neq v}}d_{uv}, \tag{7}$$

其中|S|和 S 分别表示选择的种子节点的数量和选择的初始节点集合; d_{uv} 是从节点 u 到节点 v 的最短路径的长度.

6 实 验

6.1 相关系数

在表 5 中显示了所提出的方法以及其他索引方法与 SIR 模型模拟得出的排名 R 之间的 Kendall τ 秩相关系数, 在表中加粗字体对应最优值. 从表 5 可以看出经典 DC 网络算法的性能并不太理想, 在 Sex 网络中, 虽然 IE_+算法表现不是最佳, 但与最优值对应的算法 $C_{\mathrm{nc}+}$相比相差不大, 尽管本文提出的 IE_+算法并不是在所有网络中都表现最好, 但在大多数网络中比其他算法更具有表现力.

6.2 单调关系

本文对各算法单调性进行了度量. 算法的单调性越高说明该方法在确定唯一排名的能力越强, 在表中加粗字体对应最优值. 从表 6 可以看出, 在大多数网络中, 本文提出的 IE_+算法与 IKS 方法相比

表 5 SIR 模型中节点影响指数 R 与五个中心性指数之间的 Kendall Tau

Table 5. The Kendall Tau between the node influence index R of SIR model and five centrality indices.

Network	DC	CC	C_{nc}	$C_{\mathrm{nc}+}$	ks	IKS	IE_+
NS	0.4593	0.3829	0.5604	0.7074	0.4643	0.7301	**0.8958**
EEC	0.8584	0.8238	0.8999	0.8771	0.8754	0.8963	**0.9017**
PB	0.8443	0.7956	0.8771	0.8667	0.8653	0.8859	**0.9465**
Facebook	0.6255	0.4948	0.7416	0.8614	0.6773	0.8926	**0.9364**
WV	0.8022	0.8583	0.8992	0.8939	0.9171	0.8981	**0.9661**
Sport	0.6909	0.6891	0.7875	0.8025	0.7437	0.8583	**0.9197**
Sex	0.4119	0.7329	0.7623	**0.8283**	0.5151	0.8065	0.8174
CondMat	0.5912	0.7268	0.7303	0.8114	0.6464	0.8565	**0.9254**

表 6 不同排序方法的单调性 M

Table 6. The monotonicity M of different ranking methods.

Network	M(DC)	M(CC)	$M(C_{\mathrm{nc}})$	$M(C_{\mathrm{nc}+})$	M(ks)	M(IKS)	$M(IE_+)$
NS	0.7642	**0.9927**	0.9302	0.9593	0.6428	0.8286	0.9221
EEC	0.9571	0.9828	0.9748	**0.9998**	0.9216	0.9328	0.9881
PB	0.9328	0.9301	0.9433	0.9586	0.9063	0.9266	**0.9721**
Facebook	0.9398	0.9667	0.9355	0.9646	0.9419	0.9457	**0.9898**
Sport	0.9032	0.9534	0.9292	0.9377	0.8606	0.9137	**0.9818**
Sex	0.6001	0.9122	0.9332	0.9581	0.5287	0.9248	**0.9989**
CondMat	0.8615	0.9544	0.9871	0.9864	0.8032	0.9069	**0.9996**

单调性较强. 虽然在一些网络 (如 NS, EEC 等网络) 单调性上表现不是最佳的, 但在较大网络上本文算法单调性要比其它算法单调性要高.

6.3 平均最短路径长度

为避免富人俱乐部现象带来的影响, 将排序得到的重要节点作为初始感染节点, 在传播中当节点的感染概率相等时, 为得到更广的传播范围则需要更多的初始感染节点分散在网络中. 本文进一步测试了不同算法下不同比例的初始感染节点之间的平均最短距离. 如图 2 所示, 通过不同算法排序得出的节点集合, 选取了前 2%—20% 的重要节点, 发现, 除了 EEC 网络, 随着初始感染节点的比例不断扩大, 重要节点之间的平均距离也在相应扩大. 这更进一步说明了本文提出的 IE_+算法在避免富人俱乐部现象方面具有较为优秀的表现.

6.4 重要节点性能

对节点进行排序的最终目的是为了挖掘出对传播过程起关键作用的节点, 换句话说, 如果通过排序算法得到的重要节点对传播过程起不到很好的作用, 那么这样的排序算法是不可靠的. 本小节中从不同角度来评判 IE_+算法识别的重要节点的在网络传播中的表现情况. 因在此部分讨论的是前 k 个重要节点在网络中的传播规模, 本文引入一些个经典的影响力最大化算法 (CI[44], CELF++[45], IRIE[46]) 作为比较算法.

在图 3 中, 选择网络中排名靠前的 2%—20% 个节点, 计算在感染值为 $\beta_{\mathrm{c}} = 2\beta_{\mathrm{th}}$, $\lambda = 1$, $t = 20$ 时感染的节点总数 (不包括初始感染节点) 与网络节点总数的百分比. 我们惊讶地发现在大多数网络中, 在 CC, $C_{\mathrm{nc}+}$, ks, IKS 算法下, 随着种子节点

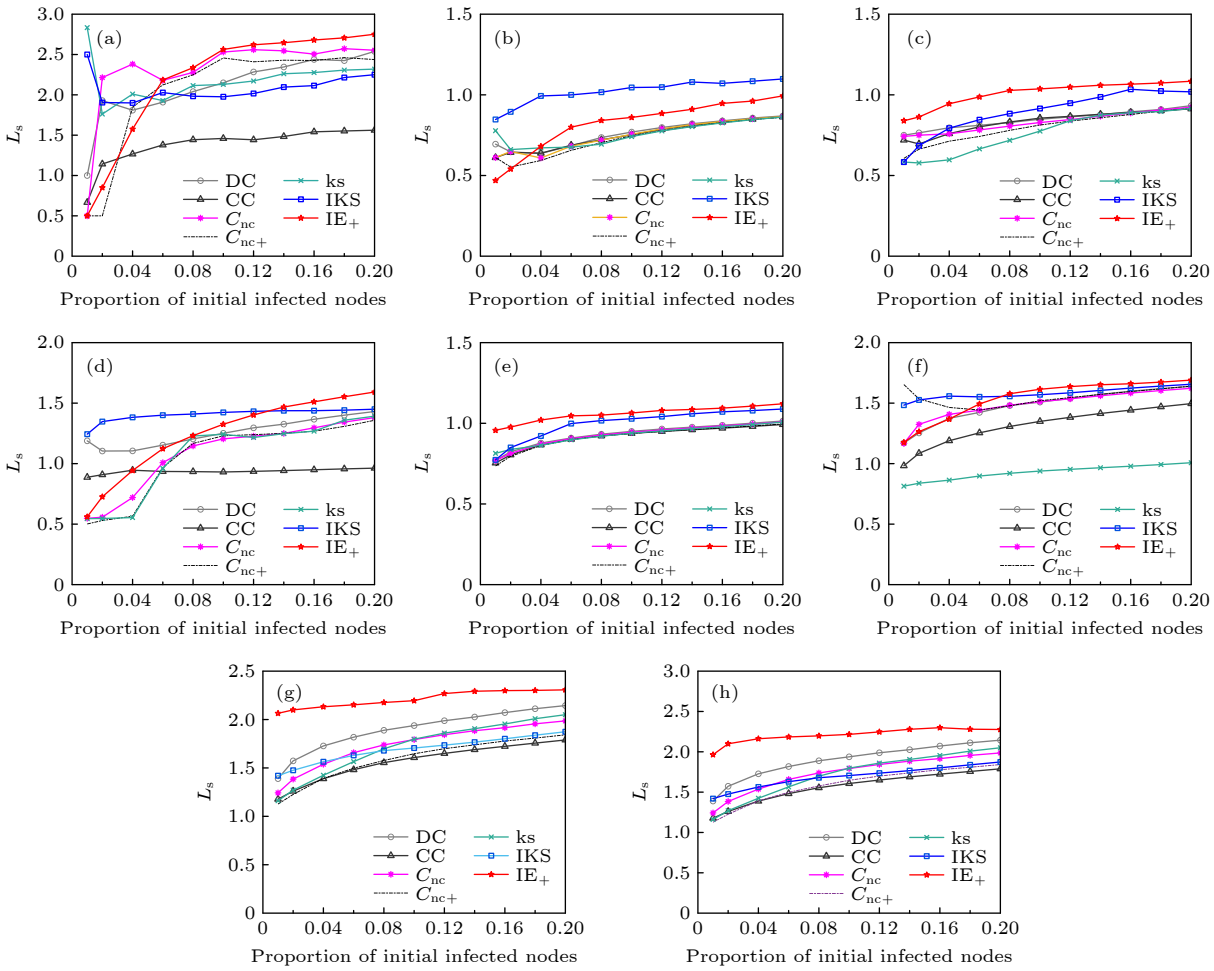

图 2 不同方法下不同比例源扩散器的平均最短路径长度 L_{s} (a) NS; (b) EEC; (c) PB; (d) Facebook; (e) WV; (f) Sport; (g) Sex; (h) CondMat

Fig. 2. Average shortest path length L_{s} under different proportion of source spreaders by different methods: (a) NS; (b) EEC; (c) PB; (d) Facebook; (e) WV; (f) Sport; (g) Sex; (h) CondMat.

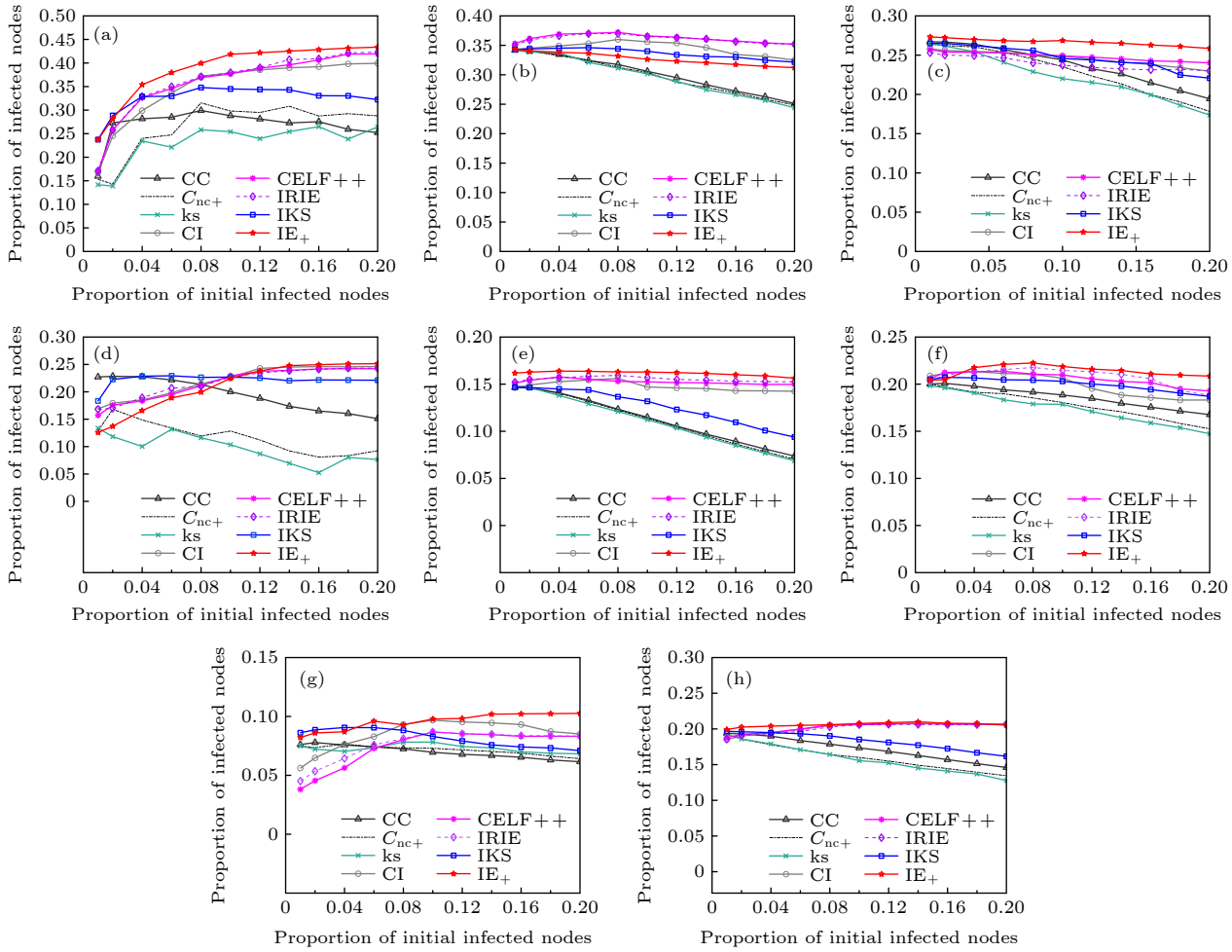

图 3 比较在相同时间内感染节点总数的百分比 (a) NS; (b) EEC; (c) PB; (d) Facebook; (e) WV; (f) Sport; (g) Sex; (h) CondMat

Fig. 3. Compare the percentage of the total number of infected nodes over the same time period: (a) NS; (b) EEC; (c) PB; (d) Facebook; (e) WV; (f) Sport; (g) Sex; (h) CondMat.

数的增加, 感染的节点总数反而在减少, 这种现象是因为随着种子节点越来越紧密地聚集在一起, 它们开始重叠, 无法再有效地传播. 而在 NS, Facebook, Sex 和 CondMat 网络中, IE_+算法随着初始感染节点总数的增加相应的感染节点总数也在增加, 在其余网络中虽然 IE_+算法随着初始感染节点的增加而略有下降, 但下降趋势相对缓慢. 除 EEC 的其余网络中, IE_+ 算法优于经典的影响力最大化算法 (CI, CELF++, IRIE), 或与其表现出相同的优势.

在六个网络中, 通过各排序算法计算后, 选择排在前 10% 的节点作为初始感染节点. 在上述相同的感染概率和恢复率下, 记录不同传播时间感染节点数与总节点数的百分比. 从图 4 可以看出, 随着时间的增加, 感染节点的数量先增加后趋于稳定, 该现象是因为随着传播时间达到一定的时长时, 网络处于稳定状态. 在 NS, Facebook, WV, Sport 和 CondMat 网络中, 本文提出的 IE_+算法具有最广泛的传播范围.

6.5 时间复杂度

在时间复杂度方面, 由于需要节点的 ks 值来计算节点的信息熵, 所以本文所提出的 IE_+算法的时间复杂度主要体现在迭代因子 IT 和 ks 值的计算上. 计算节点迭代因子 IT 和节点的 ks 值的时间复杂度都是 $O(n)$, 即本文算法的时间复杂度是 $O(n)$, IKS, C_{nc}, C_{nc+}, DC, ks 的时间复杂度都是 $O(n)$, CC 的时间复杂度为 $O(mn)$. 因此, 就时间复杂度而言, 我们的算法并不比其他算法具有更高的时间复杂度. 在相同的时间复杂度下, IE_+算法在几个考察指标上都具有良好的表现.

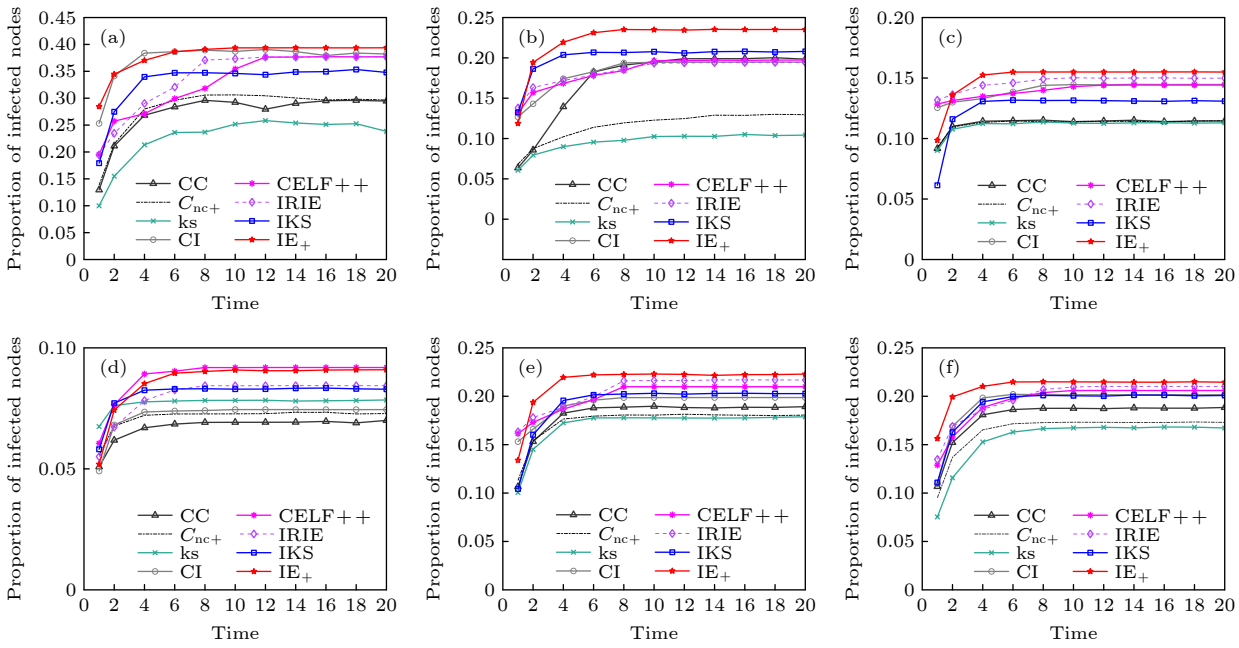

图 4 比较不同传播时间 t 中前 10% 种子节点感染节点的百分比 (a) NS; (b) Facebook; (c) WV; (d) Sex; (f) Sport; (f) CondMat

Fig. 4. Compare the percentage of nodes infected by the top 10% of seed nodes in different propagation time t: (a) NS; (b) Facebook; (c) WV; (d) Sex; (f) Sport; (f) CondMat.

7 结 论

在复杂网络分析中, 对节点的影响力进行识别和排序是一个基础性工作. 本文的研究目的是将信息熵与迭代因子相结合, 提出一种新的节点影响力评价指标, 通过排序算法得到的重要节点即使在受到富人俱乐部现象的影响下也依然具有很好的传播效果, 基于该指标, 利用迭代因子和改进的信息熵, 提出了衡量节点重要性方法 IE_+. 通过在 Kendall Tau 相关系数、单调性、平均最短距离以及节点性能上的对比实验, 表明本文提出的算法能有效对节点的重要性进行评估, 并能很好地规避富俱乐部现象, 对复杂网络中的重要节点识别工作具有较强的借鉴意义.

参考文献

[1] Pastor-Satorras R, Vespignani A 2002 *Phys. Rev. E* **65** 036104

[2] Leskovec J, Adamic L A, Huberman B A 2007 *Acm Trans. Web* **1** 5

[3] Freeman L C 1978 *Soc. Networks* **1** 215

[4] Freeman L C 1977 *Sociometry* **40** 35

[5] Sabidussi G 1966 *Psychometrika* **31** 581

[6] Lü L Y, Zhou T, Zhang Q M, Stanley H E 2016 *Nat. Commun.* **7** 10168

[7] Lü L Y, Chen D B, Ren X L, Zhang Q M, Zhang Y C, Zhou T 2016 *Phys. Rep.* **650** 1

[8] Brin S, Page L 1998 *Comput. Netw. ISDN Syst.* **30** 107

[9] Lü L Y, Zhang Y C, Yeung C H, Zhou T 2011 *PloS One* **6** 21202

[10] Kitsak M, Gallos L K, Havlin S, Liljeros F, Muchnik L, Stanley H E, Makse H A 2010 *Nat. Phys.* **6** 888

[11] Pei S, Muchnik L, Andrade J S, Zheng Z M, Makse H A 2014 *Sci. Rep.* **4** 5547

[12] Montresor A, De Pellegrini F, Miorandi D 2011 *Proceedings of the 30th Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing* San Jose, CA, June 6–8, 2011 p207

[13] Basaras P, Katsaros D, Tassiulas L 2013 *Computer* **46** 24

[14] Wang Z X, Zhao Y, Xi J K, Du C J 2016 *Physica A* **461** 171

[15] Zhou S, Mondragon R J 2004 *IEEE Commun. Lett.* **8** 180

[16] Wang M, Li W C, Guo Y N, Peng X Y, Li Y X 2020 *Physica A* **554** 124229

[17] Zareie A, Sheikhahmadi A, Jalili M, Fasaei M S K 2020 *Knowledge-Based Syst.* **194** 105580

[18] Pastor-Satorras R, Vespignani A 2001 *Phys. Rev. Lett.* **86** 3200

[19] Hethcote H W 2000 *SIAM Rev.* **42** 599

[20] Ma L I, Ma C, Zhang H F, Wang B H 2016 *Physica A* **451** 205

[21] Li Z, Ren T, Ma X Q, Liu S M, Zhang Y X, Zhou T 2019 *Sci. Rep.* **9** 8387

[22] Bae J, Kim S 2014 *Physica A* **395** 549

[23] Bhat N, Aggarwal N, Kumar S 2020 *Procedia Comput Sci.* **171** 662

[24] Ruan Y R, Lao S Y, Tang J, Bai L, Guo Y M 2020 *Acta Phys. Sin.* **71** 176401 (in Chinese) [阮逸润, 老松杨, 汤俊, 白亮 2020 物理学报 **71** 176401]

[25] Colizza V, Flammini A, Serrano M A, Vespignani A 2006 *Nat. Phys.* **2** 110

[26] Rui X B, Meng F R, Wang Z X, Yuan G 2019 *Appl. Intell.* **49** 2684
[27] Liu D, Jing Y, Zhao J, Wang W J, Song G J 2017 *Sci. Rep.* **7** 43330
[28] Namtirtha A, Dutta A, Dutta B 2018 *Physica A* **499** 310
[29] Kim H, Anderson R 2012 *Phys. Rev. E* **85** 026107
[30] Takaguchi T, Sato N, Yano K, Masuda N 2012 *New J. Phys.* **14** 093003
[31] Qu C Q, Zhan X X, Wang G H, Wu J L, Zhang Z K 2019 *Chaos* **29** 033116
[32] Hu G, Xu L P, Xu X 2021 *Acta Phys. Sin.* **70** 108901 (in Chinese) [胡钢, 许丽鹏, 徐翔 2021 物理学报 **70** 108901]
[33] Newman M E J 2006 *Phys. Rev. E* **74** 036104
[34] Yin H, Benson A R, Leskovec J, Gleich D F 2017 *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Halifax, Candana) August 13–17, 2017 p555
[35] Adamic L A 2005 *Glance N Proceedings of the 3rd International Workshop on Link Discovery* (New York, USA) 2005 p36
[36] Mcauley J, Leskovec J 2012 *Proceedings of the 25th International Conference on Neural Information Processing Systems* (Lake Tahoe, Nevada) 2012 p539
[37] Leskovec J, Huttenlocher D, Kleinberg J 2010 *Proceedings of the 19th International Conference on World Wide Web* (New York, USA) 2010 p65
[38] Rozemberczki B, Davies R, Sarkar R, Sutton C 2019 *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining* New York, USA, 2019 p65
[39] Rocha L, Liljeros F, Holme P 2011 *PLoS Comput. Biol.* **7** 1001109
[40] Leskovec J, Kleinberg J, Faloutsos C 2007 *ACM Trans. Knowl. Discovery Data* **1** 2
[41] Moreno Y, Pastor-Satorras R, Vespignani A 2002 *Eur. Phys. J. B* **26** 521
[42] Kenall M G 1938 *Biometrika* **30** 81
[43] Zhang J X, Chen D B, Dong Q, Zhao Z D 2016 *Sci. Rep.* **6** 27823
[44] Morone F, Makse H 2015 *Nature* **524** 65
[45] Goyal A, Lu W, Lakshmanan L 2011 *Proceedings of the 20th International Conference on World Wide Web* Hyderabad, India, 2011 p47
[46] Jung K, Heo W, Chen W 2012 *IEEE 12th International Conference on Data Mining* Brussels, Belgium, December 10–13, 2012 p918

Importance evaluation method of complex network nodes based on information entropy and iteration factor

Wang Ting-Ting Liang Zong-Wen † Zhang Ruo-Xi

(*School of Computer Science, Southwest Petroleum University, Chengdu* 610500, *China*)

(Received 27 September 2022; revised manuscript received 27 November 2022)

Abstract

In the study of complex networks, researchers have long focused on the identification of influencing nodes. Based on topological information, several quantitative methods of determining the importance of nodes are proposed. K-shell is an efficient way to find potentially affected nodes. However, the K-shell overemphasizes the influence of the location of the central nodebut ignores the effect of the force of the nodes located at the periphery of the network. Furthermore, the topology of real networks is complex, which makes the computation of the K-shell problem for large scale-free networks extremely difficult. In order to avoid ignoring the contribution of any node in the network to the propagation, this work proposes an improved method based on the iteration factor and information entropy to estimate the propagation capability of each layer of nodes. This method not only achieves the accuracy of node ordering, but also effectively avoids the phenomenon of rich clubs. To evaluate the performance of this method, the SIR model is used to simulate the propagation efficiency of each node, and the algorithm is compared with other algorithms. Experimental results show that this method has better performance than other methods and is suitable for large-scale networks.

Keywords: influential nodes, iteration factor, information entropy, complex networks

PACS: 89.20.Ff, 02.10.Ox, 89.75.Fb **DOI:** 10.7498/aps.72.20221878

† Corresponding author. E-mail: zongwen-liang@hotmail.com

物理学报 Acta Physica Sinica
Chinese Physical Society
Institute of Physics, CAS

基于信息熵与迭代因子的复杂网络节点重要性评价方法

汪亭亭　梁宗文　张若曦

Importance evaluation method of complex network nodes based on information entropy and iteration factor

Wang Ting-Ting　Liang Zong-Wen　Zhang Ruo-Xi

引用信息 Citation: Acta Physica Sinica, 72, 048901 (2023)　DOI: 10.7498/aps.72.20221878
CSTR: ${metaArticle.multidivStyle}
在线阅读 View online: https://doi.org/10.7498/aps.72.20221878
当期内容 View table of contents: http://wulixb.iphy.ac.cn