计算机科学
COMPUTER SCIENCE

一种基于节点稳定性和邻域相似性的社区发现算法

郑文萍，刘美麟，杨贵

引用本文

郑文萍，刘美麟，杨贵. 一种基于节点稳定性和邻域相似性的社区发现算法[J]. 计算机科学, 2022, 49(9): 83-91.

ZHENG Wen-ping, LIU Mei-lin, YANG Gui. Community Detection Algorithm Based on Node Stability and Neighbor Similarity[J]. Computer Science, 2022, 49(9): 83-91.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.220400146

一种基于节点稳定性和邻域相似性的社区发现算法

郑文萍[1,2,3]　刘美麟[1]　杨　贵[1]
1 山西大学计算机与信息技术学院　太原 030006
2 山西大学计算智能与中文信息处理教育部重点实验室　太原 030006
3 山西大学智能信息处理研究所　太原 030006

摘　要　复杂网络规模的增大导致网络中社区结构变得复杂,节点与社区之间的关系更多样化,有效度量大规模网络中节点邻域的社区构成,并对社区归属确定性有差异的节点分别进行处理,可以提高算法的社区发现质量。基于此,提出了一种基于节点稳定性和邻域相似性的社区发现算法(Node Stability and Neighbor Similarity Based Community Detection Algorithm,NSNSA)。首先定义节点的标签熵并对节点在社区发现过程中的稳定性进行度量,选择标签熵较低的节点作为稳定节点集;其次根据节点邻域的标签构成情况定义节点的邻域相似性,对节点与其邻居节点的社区归属一致性进行度量;然后利用稳定节点与其直接邻居中邻域相似性最高的节点构造初始网络,并在该子网络上运行标签传播算法,以得到可靠性较高的初始社区发现结果;最后将未聚类节点分配至与其 Katz 相似性最高的节点所在的社区,对小规模社区进行合并处理,以得到最终的社区划分结果。在真实网络及人工网络数据集上,与 LPA,BGLL,Walktrap,Infomap,LPA-S 等经典社区发现算法的对比实验表明,NSNSA 算法在模块度以及标准互信息方面表现良好。

关键词:复杂网络;社区结构;标签熵;节点稳定性;邻域相似性
中图法分类号　TP391

Community Detection Algorithm Based on Node Stability and Neighbor Similarity

ZHENG Wen-ping[1,2,3],LIU Mei-lin[1] and YANG Gui[1]
1 School of Computer & Information Technology,Shanxi University,Taiyuan 030006,China
2 Key Laboratory of Computation Intelligence and Chinese Information Processing,Ministry of Education,Shanxi University,Taiyuan 030006,China
3 Institute of Intelligent Information Processing,Shanxi University,Taiyuan 030006,China

Abstract　With the increase of the scale of complex network,community structure becomes more complex. The relationship between nodes and communities become more diversified. It is expected to improve community detection algorithm performance by effectively measuring the community structure and dealing with the nodes with different certainty of community belonging. This paper proposes a community detection algorithm based on node stability and neighbor similarity. Firstly,label entropy of nodes is defined to measure node stability and the nodes with low label entropy are selected as stable node sets. Then the neighbor similarity is defined according to the label of node neighbor and the community belonging consistency of nodes and their neighbors is measured. The initial network is constructed by using the node with the highest neighbor similarity between the stable node and its neighbor,and the initial community detection results with high reliability are obtained by running label propagation algorithm on the subnetwork. The unclustered nodes are allocated to the community of the node with the highest Katz similarity. The final result of community detection is obtained by merging small-scale communities. Compared with LPA,BGLL,Walktrap,Infomap,LPA-S and other classical algorithms,experimental results show that the NSNSA algorithm performs well in modularity and NMI.

Keywords　Complex network,Community structure,Label entropy,Node stability,Neighbor similarity

1　引言

现实世界中的复杂系统均可抽象为个体及其关系组成的复杂网络,如万维网络、引文网络等[1]。社区结构是复杂网络中研究最广泛的结构特征之一,社区通常对应着复杂系统中具有相同功能的个体集,如万维网中具有相同主题的网站

到稿日期:2022-04-14　返修日期:2022-06-03
基金项目:国家自然科学基金(62072292);山西省 1331 工程项目
This work was supported by the National Natural Science Foundation of China(62072292) and 1331 Engineering Project of Shanxi Province,China.
通信作者:郑文萍(wpzheng@sxu.edu.cn)

集合、引文网络中具有相同研究主题的文章集合等。社区结构通常呈现“低耦合、高内聚”的特点，即同一社区内部节点连接紧密，而不同社区之间节点连接稀疏。发现网络中的社区结构有助于分析网络中节点、边、子图等拓扑结构间的内在联系，发现网络动态演化规律等，从而准确理解复杂系统的拓扑结构及动力学特性[2-5]。

社区发现是一种根据网络中节点的邻域信息来寻找连接紧密的节点集的过程。研究者从各个角度提出了不同种类的社区发现算法，主要可以分为基于模块度优化的社区发现算法、基于信息传播的社区发现算法，以及基于种子扩展的社区发现算法。但现存的社区发现算法存在社区发现结果不稳定、无法应用于大规模网络等问题。对此，本文提出了一种基于节点稳定性和邻域相似性的社区发现算法。该算法首先利用传统不稳定的社区发现算法得到节点的标签分布情况，计算节点的标签熵来度量节点在社区发现过程中的稳定性，从而得到稳定节点集；然后利用节点的邻域相似性指标来度量节点的邻域相似性，选择邻域相似性高的节点来扩展稳定节点，稳定节点以及扩展节点共同构成了初始子网络，在该子网络上运行标签传播算法得到可靠性较高的初始社区发现结果；最后将未聚类节点分配至已存在的社区中，对小规模社区进行合并处理，得到最终的结果。在真实网络及人工网络数据集上，与5种经典社区发现算法进行了对比，实验结果表明，NSNSA算法在模块度以及标准互信息方面表现良好。

2 相关工作

2004年Newman等提出了模块度Modularity[6]，它是一种用于评价社区质量的度量指标。真实网络与相应零模型相差越大，对应的模块度值越高，社区结构就越合理。模块度的定义如下：

$$Q=\frac{1}{2m}\sum_{ij}(a_{ij}+\frac{d_i d_j}{2m})\delta(C_i,C_j) \tag{1}$$

其中，m 是网络的边数，$\boldsymbol{A}=(a_{ij})$ 是网络的邻接矩阵，d_i 表示节点 i 的度，C_i 表示节点 i 所属的社区，如果节点 i 和节点 j 归属于同一个社区，则 $\delta(C_i,C_j)=1$，否则 $\delta(C_i,C_j)=0$。为了便于计算模块度，提出了一种等价表示如式(2)所示：

$$Q=\sum_{c=1}^{k}\left[\frac{l_c}{m}-\left(\frac{d_c}{2m}\right)^2\right] \tag{2}$$

其中，k 是社区数，l_c 是社团 c 内部所包含的边数，d_c 是社团 c 中所有节点的度值之和。

研究者们将模块度指标作为社区发现的目标函数，通过优化基于模块度的目标函数来得到模块度较高的社区划分结果。2008年Blondel等提出的BGLL算法[7]初始时将每个节点看作一个单独的社区，然后迭代地将每个节点移动至使模块度增量最大的邻居社区中，直到模块度不再增加为止。2019年Traag等提出的Leiden算法[8]对BGLL算法进行了改进，随机选择节点的邻居社区计算模块度增益以降低算法的时间复杂度。基于模块度优化的方法通常计算代价高且存在分辨率限制的问题[9]，不适用于大规模网络中的社区发现，较难发现规模较小的社区结构。

基于信息传播的算法是通过模拟信息流在网络中的传播过程来进行社区发现的，包括基于标签传播的算法[10-12]、基于信息编码的算法Infomap[13]、基于随机游走的算法Walktrap[14]等。2007年Raghavan等提出了标签传播算法(Label Propagation Algorithm，LPA)[10]，其由于具有接近线性的时间复杂度，因此已被广泛应用于大规模复杂网络的社区发现中。LPA算法的节点更新顺序和标签更新策略中存在较大的随机性，导致其社区发现结果不稳定，即算法在同一个网络上的多次执行结果差异很大。Li等提出了一种分阶段的标签传播算法LPA-S[12]，该算法选择最相似邻居节点标签来更新当前节点标签，但由于其社区合并过程中需要更新节点相似性矩阵，时间复杂度较高，不适用于大规模网络。Zheng等[15]提出了一种两阶段的标签传播算法LPA-TS，该算法定义节点的社区参与系数以确定节点的更新顺序，依据节点间的相似性更新节点标签，从而得到最终的社区划分结果。改进的标签传播算法根据节点特征降低节点更新顺序和标签选择过程的随机性，在一定程度上提高了算法的稳定性。实际上，网络中与其他社区连接较少的节点通常具有比较明确的社区归属，多次运行后标签传播算法后这些节点的社区划分结果变化较小；与其他社区存在较多连接的节点在网络上的社区归属通常不明确，导致算法多次运行得到差异较大的社区划分结果。

基于种子扩展的社区发现算法选择社区代表性高的节点作为种子节点，根据某种适应度函数扩展种子节点得到社区发现结果。Lancichinetti等[16]提出了LFM算法，该算法初始时选择随机的种子，然后选择使社区连边密度增大的节点进行社区扩展，得到社区内部连接更紧密的社区，但由于种子节点的选择是随机的，因此会导致社区发现结果存在不稳定性。Shen等[17]对LFM算法进行了改进，将最大团作为初始种子社区，在一定程度上提高了算法的稳定性。然而，由于算法需要确定网络中的最大团，因此时间复杂度较高。选择社区代表性高的种子节点是基于种子扩展策略的社区发现算法的关键，种子节点的质量对算法稳定性和社区发现质量有很大影响。尽管度较大的节点通常与社区中的多数节点有连接，但会存在较多的社区外的连边，会影响社区发现的质量，因此大度节点并不适合作为种子节点。选择一个合适的种子节点至关重要，当选择邻域节点社区归属较为单一的节点作为种子节点时，对其进行扩展后可以一定程度地提高算法的准确性。

3 基础知识

$G=(V,E)$ 表示一个复杂网络，其中 $V=\{v_1,\cdots,v_n\}$ 表示节点集，E 表示边集，令 $n=|V|$ 且 $m=|E|$。除特别声明外，下文仅考虑无权无向简单图。图 G 的邻接矩阵用 $\boldsymbol{A}$ 来表示，其中矩阵元素 A_{ij} 定义为：

$$A_{ij}=\begin{cases}1, & (v_i,v_j)\in E\\ 0, & (v_i,v_j)\notin E\end{cases}$$

令 $N_v=\{u|(u,v)\in E\wedge u\in V\}$ 表示节点 v 在 G 中的邻居节点集合，称为 v 的直接邻域。记节点 i 的度 $d_i=|N_i|$，也可记作 $d_i=\sum_{j=1}^{n}A_{ij}$。对于图 G 的节点子集 S，将图 G 中与 S 中节点直接相邻且不在 S 中的节点集合称作 S 在 G 中的邻域，记为 $N_G(S)=\bigcup_{v\in S}N_v-S$。

对于图 $G'=(V',E')$，若 $V'\subseteq V$ 且 $E'\subseteq E$，则称 G' 是 G

的一个子图，若$|V'|=|V|$且$E'\subseteq E$，则称G'是G的一个生成子图。对于节点$u,v\in V'$，如果边$(u,v)\in E$，则$(u,v)\in E'$，称G'为节点子集V'的导出子图，记为$G[V']$，简记为$[V']$。

对于图$G=(V,E)$，称$\Omega=\{C_1,\cdots,C_k\}$为图$G$的一种社区划分，其中$C_i\subseteq V$，$C_i\neq\emptyset(1\leqslant i\leqslant k)$且$C_i\cap C_j=\emptyset(i\neq j)$。映射$f:\Omega\to\{1,\cdots,k\}$为$\Omega$中的每个社区赋予唯一的标签。对于节点$v\in C_i$，称$f(C_i)$是节点$v$在划分$\Omega$下的社区标签，记作$l_\Omega(v)=f(C_i)$。对于节点子集$S\subseteq V$，称$l_\Omega(S)=\{l_\Omega(v)\mid v\in S\}$为子集$S$对划分$\Omega$的社区标签集。

4 一种基于节点稳定性和邻域相似性的社区发现算法 NSNSA

NSNSA 算法包括 5 个主要的过程：1)提出了一种节点稳定性度量指标，以度量网络中节点的稳定性；2)利用邻域相似性指标对稳定节点进行扩展；3)从原网络中抽取稳定节点和扩展节点的子网络，得到初始社区划分结果；4)将未聚类节点进行分配；5)社区合并。

4.1 节点稳定性度量指标

随着网络规模的增大，节点和社区之间的关系变得复杂，如果一个节点的邻居节点分布于多个社区，则表明该节点的社区归属尚不明确，社区划分结果不稳定；反之，如果一个节点的邻居均在一个社区，则该节点的社区归属明确，社区划分结果较为稳定。

香农熵通常被用于度量信息的不确定性程度，熵值越大，信息的不确定性程度就越高；熵值越小，信息的确定性程度就越高。基于此，我们利用香农熵的形式来度量节点的邻居节点社区归属的不确定性程度，进而度量节点在社区划分过程中的稳定性，称之为节点v在社区划分Ω下的标签熵。

定义 1(标签熵)　假设$\Omega=\{C_1,\cdots,C_k\}$为图$G$的一种社区划分，其标签映射为$f:\Omega\to\{1,\cdots,k\}$。若节点$v$的邻域$N_v$中包含$k_v$个不同的社区标签，记作$l_\Omega(N_v)=\{l_1,\cdots,l_{k_v}\}$，则称$k_v$为节点$v$邻域内的标签类别数。记$S_v(l_i)=\{u\mid l_\Omega(u)=l_i,u\in N_v\}$为$N_v$中标签为$l_i$的节点集合，令$s_v(l_i)=|S_v(l_i)|$。将节点$v$在划分$\Omega$下的标签分布定义为：

$$P_\Omega(v)=\left\{\frac{s_v(l_1)}{|N_v|},\cdots,\frac{s_v(l_{k_v})}{|N_v|}\right\} \tag{3}$$

则定义节点v在划分Ω下的标签熵为：

$$H_\Omega(v)=-\sum_{i=1}^{k_v}\frac{s_v(l_i)}{|N_v|}\log\left(\frac{s_v(l_i)}{|N_v|}\right) \tag{4}$$

若$|N_v|=0$，则令$H_\Omega(v)=0$。

节点的标签熵可以用于评价在社区划分Ω下网络中节点邻域社区分布的确定性。节点v的标签熵$H_\Omega(v)$越小，该节点邻域内的节点社区归属越一致，节点v在社区划分过程中的表现越稳定；反之，节点v的标签熵$H_\Omega(v)$越大，该节点的邻居节点的社区归属越混乱，节点v越不稳定。利用标签熵可以区分节点在社区划分过程中的稳定性，先确定稳定节点的社区归属可以提高算法的稳定性。

图 1 给出了 Dolphins 网络的真实社区划分结果，其中节点 28 的度$d_{28}=5$，其邻域中节点分散到两个社区C_1和C_2中，标签分别为l_1和l_2，即$l_\Omega(N_{28})=\{l_1,l_2\}$且$s_{28}(l_1)=1$，$s_{28}(l_2)=4$，则节点 28 在该划分下的标签分布$P_\Omega(28)=\{0.2,0.8\}$，对应标签熵$H_\Omega(28)=0.72$。而对于节点 3，其邻居节点都属于社区$C_2$，因此在该划分下的标签熵$H_\Omega(3)=0$，节点 28 比节点 3 具有更高的不稳定性。实际上，位于两个社区边界的节点具有较高的标签熵，如节点{1,7,19,28,30,36,39,40,57}，在社区发现过程中这些节点的社区归属往往是不稳定的。

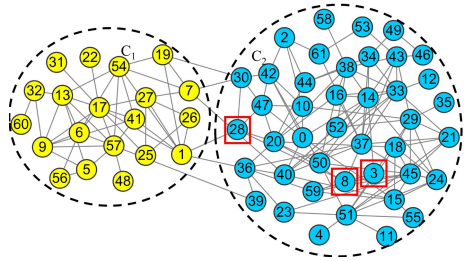

图 1　Dolphins 网络的真实社区划分结果

Fig. 1　Real community partition result of Dolphins network

根据式(4)选择稳定节点，但由于稳定节点较少且在网络中分布较为分散，无法直接抽取由稳定节点构成的子图，因此需要扩展一部分节点。我们扩展稳定节点的邻居节点，但部分节点的邻域社区构成比较复杂。如图 1 所示，节点 57 的邻居节点为$N_{57}=\{5,6,9,13,17,39,41,48,54\}$，如果直接将节点 57 的所有邻居节点均进行扩展，节点 39 则会出现在扩展子图中，然而节点 39 分别与社区C_1和社区C_2各连一条边，属于边界节点，若将节点 39 作为扩展节点，所抽取的子图的稳定性会有所降低，因此需要对稳定节点的邻域进行有效选择。基于此，本文提出了一种对稳定节点及其邻居节点的社区归属一致性进行度量的方法。

4.2 邻域相似性度量指标

定义 2(邻域相似性)　节点u表示得到的稳定节点，N_u表示节点u的邻域节点集，y_u表示节点u的标签，将节点u邻居节点中标签相同的节点数与整个网络中与其标签相同的节点数的比例定义为节点的邻域相似性。节点u在某种社区划分下的邻域相似性度量指标如下：

$$Score_{\mathrm{sim}}(u)=\frac{|\{v\in N_u\mid y_v=y_u\}|}{|N_u|\cdot|\{v\in V\mid y_v=y_u\}|} \tag{5}$$

节点u与它的邻居越相似，邻域相似性$Score_{\mathrm{sim}}(u)$就越高。在稳定节点扩展阶段，尽可能地选择与稳定节点社区归属一致性高的节点来进行扩展，这样会得到相对稳定的初始社区划分。而邻域相似性越高代表该节点与稳定节点的社区归属更一致。因此，邻域相似性指标能很好地刻画稳定节点与其邻居社区归属的一致性程度。

如图 1 所示，节点 8 的社区标签为l_2，其邻域节点集为$N_8=\{3,20,28,37,45,59\}$。对于节点 28，其邻居节点的标签有l_1和l_2两种，标签为l_2的节点个数为 4，整个网络中标签为l_2的节点个数为 42，则$Score_{\mathrm{sim}}(28)=4/(5\times42)=0.019$。对于节点 3，邻居节点的标签为$l_2$的节点个数为 3，则$Score_{\mathrm{sim}}(3)=3/(3\times42)=0.0238$。可以看出，8 号节点为稳定节点时，节点 3 相比节点 28 与其更相似。因此，节点 3 作为节点 8 的扩展节点更合理，所得到的扩展子图的社区发现结果更稳定。

4.3 算法过程

4.3.1 初始稳定节点选择

由于网络规模的增大，社区结构变得复杂，很多社区发现算法表现出了较强的不稳定性，即同一个网络上运行同一种社区发现算法，会得到较大差异的社区划分结果。但是，这些不同的社区划分结果可以为社区形成过程中节点的稳定性提供指导。通常具有较明确社区归属的节点表现得较为稳定，先确定这部分节点的社区归属可以有效提高算法的性能。

算法 NSNSA 首先在网络上运行 t 次 LPA 算法，得到 t 种社区划分结果 $\Omega_1,\cdots,\Omega_t$；然后分别计算每个节点 v 在 t 种划分下的标签熵 $H_{\Omega_i}(v)$，选择 t 次结果中最大的标签熵值作为该节点的熵值，最后选择熵小于一定阈值的节点作为稳定节点，网络的稳定节点集为：

$$S=\{v\mid \max_{1\leqslant i\leqslant t}(H_{\Omega_i}(v))\leqslant \varepsilon, v\in G\} \tag{6}$$

其中，ε 称作稳定性阈值，t 取值为 3。通常，网络中节点的标签熵服从幂律分布，往往仅有少量节点标签熵低于均值，因此本文选择所有节点标签熵的平均值作为阈值，将标签熵低于均值的节点确定为稳定节点。在实际中，也可以根据实际应用对不同网络设置不同的阈值。

本文以 Dolphins 网络为例展示稳定节点的选择过程。算法首先运行 3 次 LPA 算法，根据得到的社区划分结果分别计算各个节点的标签熵，选择 3 次标签熵中的最大值作为最终的熵值。表 1 列出了 Dolphins 网络的 3 次熵值结果以及最终选择的熵值。此例中计算得到的稳定性阈值 $\varepsilon=0.721$。选择熵小于稳定阈值的节点作为稳定节点集 S，得到的结果如图 2 中的蓝色节点所示。

表 1 Dolphins 网络节点的标签熵

Table 1 Label entropy of nodes in Dolphins network

v	$H_{\Omega_1}(v)$	$H_{\Omega_2}(v)$	$H_{\Omega_3}(v)$	$H_{\Omega}(v)$	v	$H_{\Omega_1}(v)$	$H_{\Omega_2}(v)$	$H_{\Omega_3}(v)$	$H_{\Omega}(v)$
0	1	1	1.4591	1.4591	31	0	0	0	0
1	1.0612	1.0612	1.9056	1.9056	32	0	0	0	0
2	1.5	1.5	1	1.5	33	0	0	0.4689	0.4689
3	0	0	0	0	34	0	0.7219	0.7219	0.7219
4	0	0	0	0	35	0	0	0	0
5	0	0	0	0	36	0.8631	0.8631	1.3787	1.3787
6	0	0	0	0	37	0.4394	0.4394	0.6840	0.6840
7	1.3709	1.3709	1.9219	1.9219	38	0	0	0	0
8	0.6500	0.6500	0.6500	0.6500	39	1	1	1	1
9	0	0	0	0	40	1.0612	1.0612	1.5487	1.5487
10	0.7219	0.7219	0.7219	0.7219	41	0	0	0	0
11	0	0	0	0	42	0.6500	0.6500	1.2516	1.2516
12	0	0	0	0	43	0.5916	1.1488	1.1488	1.1488
13	0	0	0	0	44	0.8112	0.8112	0.8112	0.8112
14	0.4138	0.4138	0.8166	0.8166	45	0	0	0.9456	0.9456
15	0.5916	0.5916	1.3787	1.3787	46	0	1	1	1
16	0	0	0	0	47	0.6500	0.6500	1.4591	1.4591
17	0	0	0.7642	0.7642	48	0	0	0	0
18	0	0	0.5916	0.5916	49	0	1	1	1
19	0.8112	0.8112	1	1	50	0.5916	0.5916	1.3787	1.3787
20	0.7642	0.7642	0.9864	0.9864	51	0	0	0.4689	0.4689
21	0	0	0.9182	0.9182	52	0	0	0.8112	0.8112
22	0	0	0	0	53	1	1	0	1
23	0	0	0.9182	0.9182	54	0	0	0.8631	0.8631
24	0	0	0.65	0.65	55	0	0	0	0
25	0	0	0.9182	0.9182	56	0	0	0	0
26	0	0	0.9182	0.9182	57	0	0	0	0
27	0	0	1.5219	1.5219	58	0	0	0	0
28	1.5219	1.5219	1.9219	1.9219	59	0	0	0.9709	0.9709
29	0.5032	0.5032	1.2243	1.2243	60	0	0	0	0
30	0.9709	0.9709	1.5219	1.5219	61	1.5849	1.5849	0.9182	1.5849

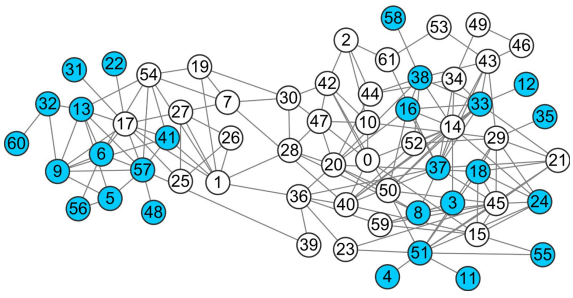

图 2 Dolphins 网络上得到的稳定节点集 S(电子版为彩图)

Fig. 2 Stable node set S in Dolphins network

4.3.2 稳定节点扩展

直接导出由稳定节点构成的子图通常存在多个连通分支，无法直接进行社区发现，需要扩展一部分节点来保证抽取子图的连通性。而复杂网络中节点的连接差异较大，存在一部分节点与网络中的大多数节点都有连边的情况，因此需要对扩展节点进行有效选择。为了保证连通性，本文从稳定节点的邻域中选择扩展节点。尽可能在邻域中选择与稳定节点社区归属更一致的节点进行扩展，从而得到更为稳定的初始社区划分结果。

采用邻域相似性式(5)对稳定节点进行扩展，选择稳定节点的邻域节点中邻域相似性最高的节点作为扩展节点，从原

网络中抽取包含稳定节点及其扩展节点的子图，在该子图上进行初始社区发现，得到最初的社区划分结果。表 2 列出了 Dolphins 网络在某种社区划分下节点的邻域相似性。表 3 列出了稳定节点的邻居节点以及邻域相似性最高的扩展结点。图 3 中绿色节点为稳定节点的扩展节点，绿色节点和蓝色节点共同构成子图 Gs。

表 2 Dolphins 网络节点的邻域相似性

Table 2 Neighborhood similarity of nodes in Dolphin network

v	$Score$	v	$Score$	v	$Score$	v	$Score$	v	$Score$
0	0.1000	1	0.0234	2	0.1000	3	0.0476	4	0.0833
5	0.0625	6	0.0625	7	0.1333	8	0.0396	9	0.0625
10	0.1600	11	0.0833	12	0.0476	13	0.0625	14	0.0396
15	0.0476	16	0.0476	17	0.0486	18	0.0714	19	0.1666
20	0.0370	21	0.0555	22	0.0625	23	0.0555	24	0.0694
25	0.2222	26	0.2222	27	0.1333	28	0.0190	29	0.0555
30	0.1333	31	0.0625	32	0.0625	33	0.0428	34	0.0380
35	0.0833	36	0.0272	37	0.0389	38	0.0476	39	0.0312
40	0.0297	41	0.0625	42	0.1333	43	0.0340	44	0.0357
45	0.0530	46	0.2500	47	0.1000	48	0.0625	49	0.2500
50	0.0272	51	0.0750	52	0.0357	53	0.0476	54	0.0446
55	0.0833	56	0.0625	57	0.0625	58	0.0476	59	0.0285
60	0.0625	61	0.0317						

表 3 Dolphins 网络稳定节点的扩展节点选择

Table 3 Extended node selection for stable nodes in Dolphin network

v	Nv	扩展节点	v	Nv	扩展节点
3	8,14,59	8	4	51	51
5	9,13,56,57	9	6	9,13,17,54,56,57	9
8	3,20,28,37,45,59	45	9	5,6,13,17,32,41,57	5
11	51	51	12	33	33
13	5,6,9,17,32,41,54,57	5	16	14,20,33,37,38,50	38
18	15,20,21,24,29,45,51	24	22	17	17
24	14,15,18,29,45,51	51	31	17	17
32	9,13,60	9	33	12,14,16,21,34,37,38,40,43,50	21
35	29	29	37	8,14,16,21,33,34,36,37,40,45,61	21
38	14,16,20,33,43,44,52,58	16	41	1,9,13,54,57	9
48	57	57	51	4,11,18,21,23,24,29,45,50,55	4
55	15,51	51	56	5,6	5
57	5,6,9,13,17,39,41,48,54	5	58	38	38
60	32	32			

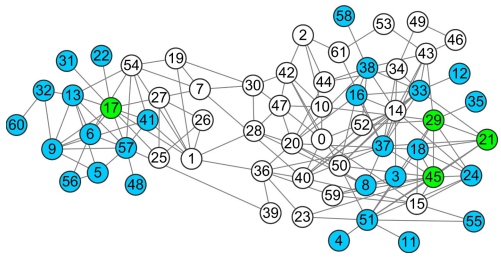

图 3 包含稳定节点集 S 及扩展节点的子图 G_S（电子版为彩图）

Fig. 3 Subgraph G_S containing stable node set S and extended nodes

4.3.3 初始社区获取

由稳定节点集 S 和扩展节点构造的子图 G_S 中包含相对稳定的节点，因此在 G_S 上进行初始社区发现可以得到较为稳定的社区划分结果。此外，由于子图中的节点较为稳定，利用不太稳定的社区发现算法进行初始社区发现，也可以得到较好的社区划分结果。因此，本文采用 LPA 算法进行初始社区划分。由于所抽子图 G_S 的节点数不多，也可以使用复杂度略高但准确度高的社区发现算法进行社区划分。

在本例中，从原始网络中抽取包含稳定节点和扩展节点共 31 个节点的子图，在该子图上运行标签传播算法得到的初始社区划分结果如图 4 中带颜色的节点所示，每个颜色代表一个社区，共 4 个社区。

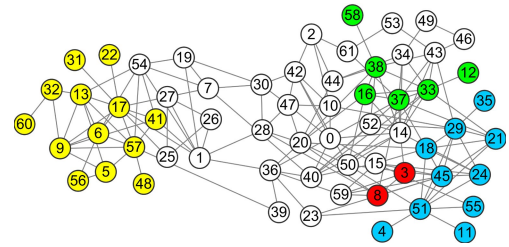

图 4 Dolphins 网络初始社区划分结果（电子版为彩图）

Fig. 4 Initial community partition result of Dolphins network

4.3.4 未聚类节点处理

得到子图中的初始社区划分之后，网络中还存在部分未聚类节点，需要对此类节点进行处理。本文利用节点的相似性，将未聚类节点划分至与其相似性最高的社区中，从而得到最终的社区划分结果。

本文采用以下 6 种相似性指标在真实带标签网络中进行评估。其中，CN，AA，CAR，HDI，HPI 为结构等价指标，而 Katz 为规则等价指标。各个指标的具体定义如下：

(1)共同邻居(CN) $S_{xy}^{CN}=|N_x \cap N_y|$;

(2) Adamic-Adar 指标(AA) $S_{xy}^{AA}=\sum_{z\in N_x\cap N_y}\frac{1}{\log d_z}$;

(3) CAR 指标 $S_{xy}^{CAR}=\sum_{z\in N_x\cap N_y}\frac{|N_x\cap N_y\cap N_z|}{|N_z|}$;

(4)大度节点不利指标(HDI) $S_{xy}^{HDI}=\frac{|N_x\cap N_y|}{\max\{d_x,d_y\}}$;

(5)大度节点有利指标(HPI) $S_{xy}^{HPI}=\frac{|N_x\cap N_y|}{\min\{d_x,d_y\}}$;

(6) Katz 指标 $S_{ij}^{Katz}=\sum_{l=1}^{\infty}\beta^l(A^l)_{ij}$。

图 4 中白色节点为未聚类节点,利用 Katz 相似性指标将 31 个未聚类节点分配至与 Katz 相似性最高的节点所在的初始社区中,从而得到社区划分结果。图 5 给出了对未聚类节点处理后所得到的社区划分结果,其中每个颜色代表一个社区,共包含 4 个社区。此时该划分结果的模块度值为 0.4556,可以看出与真实社区划分结果相近。

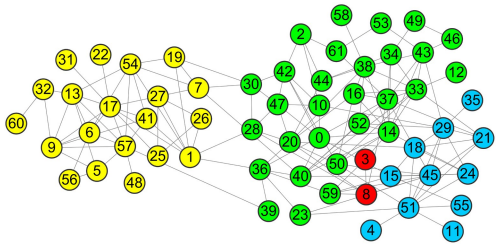

图 5 相似性分配后的社区划分结果

Fig. 5 Community partition result after similarity allocation

4.3.5 社区合并

由于对稳定节点进行邻域扩展时所选的邻居节点较少,所构造的子图可能是不连通的,未聚类节点根据相似性分配得到社区时会存在规模较小的社区,因此需要将小规模社区进行合并。本节根据模块度进行合并,将小规模社区合并至使模块度增量最大的社区中。

图 5 中,节点 3 和节点 8 被单独分为一个社区,此社区规模较小,需要进行合并操作。将红色社区合并到绿色社区后模块度值增加,并且合并之后的社区结构更明显。合并之后的最终社区发现结果如图 6 所示,对真实社区划分结果中较大规模的社区进行了更细的划分。

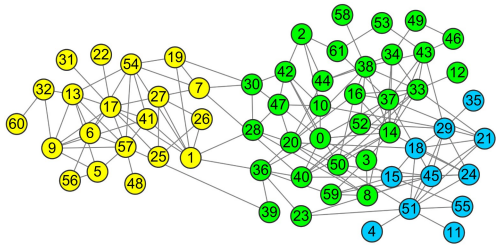

图 6 Dolphins 网络最终社区划分结果(电子版为彩图)

Fig. 6 Final community partition result of Dolphins network

4.3.6 算法框架

算法 1 给出了本文提出的基于节点稳定性和邻域相似性的社区发现算法的框架。

算法 1 基于节点稳定性和邻域相似性的社区发现算法

输入:图 $G=(V,E)$,其中 $V=\{v_1,v_2,\cdots,v_n\}$

输出:社区划分结果 $\Omega=\{C_1,\cdots,C_k\}$

1. 首先获取网络中的稳定节点集 S 以及不稳定节点集 V-S。
2. 计算每个节点的邻域相似性,选择稳定节点的邻居中邻域相似性最高的节点作为扩展节点。
3. 从原网络 G 中抽取一个包含稳定节点集 S 及扩展节点的连通子图 G_S,并对 G_S 的节点进行聚类,得到初始社区 $\Omega_S=\{C_1^S,\cdots,C_k^S\}$。
4. 根据相似性度量指标对集合 $V(G)-V(G_S)$ 中的节点聚类,得到最终的聚类结果 $\Omega=\{C_1,\cdots,C_k\}$。
5. 根据模块度进行社区合并。
6. 结束。

4.4 时间复杂度分析

对于一个包含 n 个节点、m 条边的网络 G,算法 NSNSA 首先计算节点标签熵的总代价为 $O(tm)$;接着对节点的熵进行排序,选择小于阈值的节点作为稳定节点集 S,代价为 $O(n\log n)$;因此选择网络稳定节点集的总代价为 $O(tm+n\log n)$;对稳定节点的邻域进行选择,总代价为 $O(m)$;抽取子图并聚类得到初始社区总代价为 $O(m)$;处理未聚类节点的时间代价为 $O(n^2)$;若社区个数为 k,则合并社区总代价为 $O(k^2)$。

综上所述,本文提出的算法 NSNSA 的总时间复杂度为 $O(tm+n\log n+n^2+k^2)$。由于 $k^2\ll n^2$,因此时间复杂度近似为 $O(n^2)$。

5 实验结果与分析

本节为了说明规则等价和结构等价两类相似性指标的表现情况,首先介绍了不同相似性指标下实验的比较结果,接着分别在人工网络和真实网络数据集上与经典的社区发现算法进行比较,以评估所提 NSNSA 算法的性能。

5.1 评价指标

对于没有真实标签的数据集,本文选用 Newman 提出的模块度对聚类结果进行评价,如式(2)所示,模块度越高代表结果越好。对于有标签的数据集,本文选用标准互信息(NMI)对聚类结果进行评价,NMI 越高,算法的划分结果与真实划分越一致。为了评估算法的稳定性,我们计算了 20 次实验结果模块度值的方差,方差越低,说明算法越稳定。

社区划分结果为 $\Omega=\{V_1,V_2,\cdots,V_k\}$,真实社区划分结果为 $O=\{O_1,O_2,\cdots,O_{k'}\}$。对于集合 V_i 和 O_j($1\leqslant i\leqslant k$,$1\leqslant j\leqslant k'$),令 $T_{i,j}=|V_i\cap O_j|$,$b_i=\sum_{i=1}^{k'}T_{i,j}$,$s_j=\sum_{i=1}^{k}T_{i,j}$。标准互信息 NMI 如式(7)所示:

$$NMI=\frac{2\sum_{i=1}^{k}\sum_{j=1}^{k'}T_{i,j}\log\frac{nT_{i,j}}{b_is_j}}{-\sum_{i=1}^{k}b_i\log\frac{b_i}{n}-\sum_{j=1}^{k'}s_j\log\frac{s_j}{n}} \tag{7}$$

方差用于评估算法的稳定性,如式(8)所示:

$$\sigma^2=\frac{\sum(X-\mu)^2}{N} \tag{8}$$

其中,σ^2 为总体方差,X 为变量,μ 为总体均值,N 为样本总数。

5.2 不同相似性指标实验结果

本小节介绍了不同的相似性度量指标对未聚类节点进行社区分配得到的社区划分结果的比较情况。图 7—图 9 给出

了在 Karate[18]，Dolphins[19] 和 Polbooks 网络上使用 6 种相似性指标所得的实验结果。图中，横坐标表示 15 次实验的结果，纵坐标表示 NMI 值，其中不同颜色的线代表不同指标下的实验结果，红色表示 Katz 指标，各个网络 Katz 指标的表现最好。图 10 也描述了不同网络下 NMI 的均值，说明了规则等价比结构等价考虑得更充分。

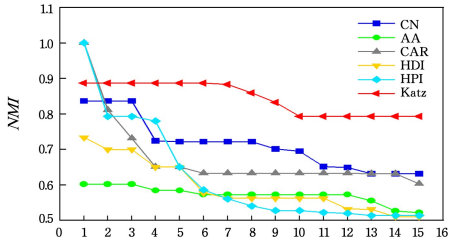

图 7 Karate 网络的实验结果(电子版为彩图)

Fig. 7 Experimental results of Karate network

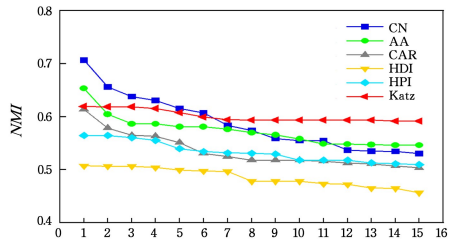

图 8 Dolphins 网络的实验结果(电子版为彩图)

Fig. 8 Experimental results of Dolphins network

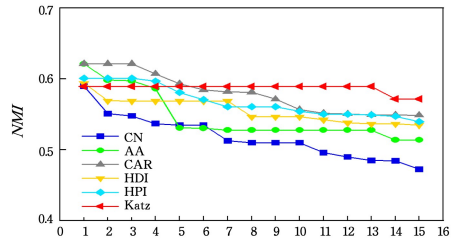

图 9 Polbooks 网络的实验结果(电子版为彩图)

Fig. 9 Experimental results of Polbooks network

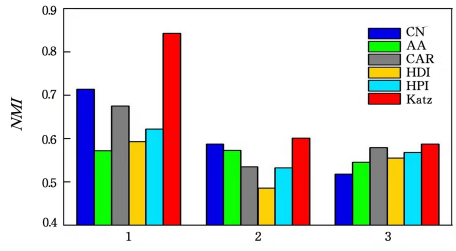

图 10 各网络均值的实验结果

Fig. 10 Experimental results of all network mean values

5.3 人工网络实验结果

本小节在 Lancichinetti 等开发的 LFR benchmark 数据集上进行算法性能的评估。相比其他人工网络数据集，LFR 网络中节点的度、社区规模等均为可调节参数，更能够反映真实网络的特性。具体来说，LFR 网络规模分别为 1000 或 5000，社区规模为 20～50，混合参数 μ 为 0.1～0.6，网络的平均度为 20，最大度为 50，节点度序列满足指数为 3 的幂律分布，社区规模系列满足指数为 1.5 的幂律分布。具体实验结果如图 11 和图 12 所示。

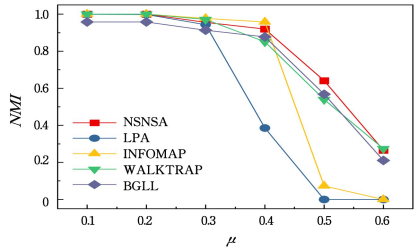

图 11 $N=1000$ LFR benchmark 网络上的实验结果

Fig. 11 Results on LFR benchmark network when $N=1000$

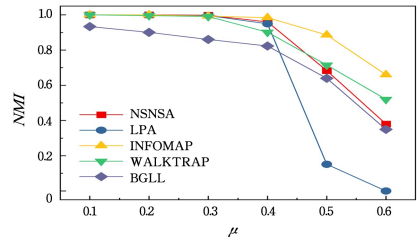

图 12 $N=5000$ LFR benchmark 网络上的实验结果

Fig. 12 Results on LFR Benchmark network when $N=5000$

由图 11 和图 12 可以看出，随着 μ 值的增加，算法性能均呈下降趋势，这是由于随着 μ 值的增大，社区向外的连边数变多，社区结构变得混乱，利用社区发现算法发现社区的能力变差。在 $N=1000$ 的 LFR 网络上，NSNSA 算法相比其他算法性能相当或表现较好。在 $N=5000$ 的 LFR 网络上，NSNSA 算法优于 LPA 和 BGLL 算法，生成的 LFR 网络社区分布较为平衡，社区结构较为明显。而 Infomap 和 Walktrap 算法是基于信息传播的方法，因此本文算法的性能相比这两种算法来说表现略差。

5.4 真实网络实验结果

本小节评估了 6 种社区发现算法在空手道俱乐部网络(Karate Club Network)、海豚社交网络(Dolphins Social Network)、Polbooks 网络、大学生足球网络(American College Football Network)[20] 和政治博客网络(Polblogs)、Yeast、NetScience、PGP、Douban 等 14 个真实网络上的实验性能。数据的基本情况如表 4 所列。

表 4 真实网络数据集

Table 4 Real network datasets

Dataset	Number of nodes	Number of edges
Karate	34	78
Dolphins	62	159
Les	77	254
Polbooks	105	441
Football	115	613
NetScience	379	914
Celegan	453	2025
Email	1133	5451
Polblogs	1490	16718
Yeast	2375	11693
Facebook	2888	2981
Power	4941	6594
PGP	10680	24316
Douban	154908	327162

表5列出了5种算法的NMI值和模块度值的实验比较结果。NSA算法为不使用邻域相似性指标，直接抽取由稳定节点及其邻居构建的子图所得的实验结果。表5中，粗体表示最好的实验结果，最后一行给出了该算法在所有网络上的平均值。可以看出，本文算法在大多数情况下得到了较好的结果。对于Polblogs网络而言，该网络是不连通的，因此在邻域扩展阶段NSA算法相比NSNSA算法会扩展更多的邻居节点来构造子网络，其结果优于NSNSA算法，但在别的网络上NSNSA算法的性能均优于NSA算法。可以看出，对稳定节点的邻居节点进行有效筛选一定程度上可以提高算法的性能。LPA-S算法在小规模网络上表现较好，但是该算法的时间复杂度较高，不适合用在大规模网络上。BGLL算法是基于模块度优化的算法，因此对于模块度指标而言，其表现良好。Infomap算法在某些网络上与本文算法的性能接近，但是从表格的最后一行可以看出，本文算法的整体均值都高于其他对比算法。

LPA算法具有近线性的时间复杂度，但实验结果表现出了较强的不稳定性，而NSNSA算法利用LPA算法的不稳定性来确定节点邻域的分布情况，从而确定稳定节点。Waltrap算法的计算代价较高，BGLL算法受模块度的精确度限制，因此在较小规模网络上表现良好，但当节点数达到十几万时(如Douban)，无法计算出实验结果。LPA-S算法由于涉及矩阵运算，因此当节点数规模较大时也无法计算。Infomap算法的运行速度较快，但是从表5可以看出，本文提出的NSNSA算法的性能优于Infomap算法。

表5 真实网络实验结果对比

Table 5 Experimental results on real networks

		LPA	Walktrap	Infomap	BGLL	LPA-S	NSA	NSNSA
Karatre	Q	0.3147	0.3532	0.4020	**0.4074**	0.369	0.3541	0.3744
	NMI	0.6296	0.5041	0.6994	0.6021	**0.9426**	0.7209	0.8864
Dolphins	Q	0.4920	0.4888	0.5158	**0.5202**	0.4373	0.4532	0.5126
	NMI	0.5259	0.5372	0.5563	0.5162	0.6820	0.6994	**0.7059**
Polbooks	Q	0.3801	0.5069	**0.5267**	0.5210	0.4971	0.4674	0.5162
	NMI	0.5186	0.5427	0.4934	0.5219	0.5543	0.5955	**0.6209**
Football	Q	0.5819	0.6029	0.5902	**0.6037**	0.5291	—	0.6046
	NMI	0.8725	0.8873	0.8801	0.8740	0.8406	—	**0.8919**
Polblogs	Q	0.3921	0.3477	0.0301	0.3361	0.3877	**0.4260**	0.4245
	NMI	0.3046	0.3230	0.1648	0.3125	0.2893	**0.3405**	0.3371
Les	Q	0.5025	0.4024	0.4427	0.4486	0.4458	0.5168	**0.5439**
NetScience	Q	0.7449	0.8146	0.7892	0.8255	0.7537	0.7873	**0.8381**
Email	Q	0.3492	0.4138	0.4798	0.4892	0.3458	0.4555	**0.5267**
Yeast	Q	0.6639	0.6445	0.5269	**0.7291**	0.5723	0.5897	0.6955
Celegan	Q	0.1310	0.1297	0.2831	0.3183	0.2415	0.3228	**0.3669**
Facebook	Q	0.7943	0.6217	0.7900	**0.8047**	0.3478	0.7937	0.7698
Power	Q	0.5954	0.8191	0.8152	**0.9309**	—	0.6205	0.6361
PGP	Q	0.7314	0.7758	0.7968	**0.8695**	0.4324	0.7406	0.7125
Douban	Q	0.4431	—	0.4871	—	—	0.4634	**0.5361**
Average		0.5246	0.5397	0.5405	0.5906	0.5099	0.5498	**0.6053**

5.5 稳定性实验结果

表6列出了NSNSA算法与LPA和LPA-S等稳定性较差的算法的实验结果。可以看出，本文算法NSNSA在大多数网络上可以得到更稳定的实验结果。

表6 稳定性实验结果对比

Table 6 Comparison of experimental results on stability

	Karate	Dolphins	Les	Polbooks	NetScience	Celegan	Email	Polblogs	Facebook	Power	PGP	Douban
LPA	0.00229	0.00112	0.00259	0.00069	0.00021	0.00185	0.01806	0.01562	**0.00000**	0.00002	0.00008	0.00269
LPA-S	**0.00009**	0.00586	0.00048	0.00041	0.00424	0.00827	**0.00040**	—	—	—	—	—
NSNSA	0.00066	**0.00046**	**0.00015**	**0.00016**	**0.00013**	**0.00158**	0.00051	**0.00000**	**0.00000**	**0.00000**	**0.00005**	**0.00000**

结束语 本文提出了一种基于节点稳定性和邻域相似性的社区发现算法NSNSA，包括稳定节点选择、稳定节点扩展、初始社区获取、未聚类节点处理、社区合并5个主要过程。算法首先提出了一种节点稳定性度量指标，称为标签熵，用于确定稳定节点集；接着对节点邻域进行有效的区分，提出邻域相似性度量指标，用于扩展稳定节点集，进而从原网络中抽取稳定子网络；然后对稳定子网络进行初始社区划分，得到初始划分结果，将未聚类节点划分至初始社区中；最后进行社区合并，得到最终的社区划分结果。在人工网络和真实网络数据集上的实验结果表明了所提NSNSA算法的优越性。结果表明，对节点的稳定性进行有效度量，优先确定稳定节点的社区归属在社区发现中是一种有效的方法。在未来，应该进一步探索在大规模网络背景下如何度量节点的稳定性。此外，本文只关注非重叠的社区发现。重叠社区发现在大规模网络中普遍存在，未来需要进一步研究在重叠社区背景下如何度量节点稳定性，从而进行重叠社区发现算法的研究工作。

参 考 文 献

[1] NEWMAN M E J. Networks: An Introduction[M]. Oxford University Press, 2010.

[2] JIA S W, GAO L, GAO Y, et al. Exploring triad-rich substructures by graph-theoretic characterizations in complex networks [J]. Physica A: Statistical Mechanics and its Applications, 2017, 468(2017): 53-69.

[3] HIDEHIKO I, MINEICHI K, ATSUYOSHI N. Partitioning of web graphs by community topology[C]// Proceedings of the 14th International Conference on World Wide. New York, ACM Press, 2005: 661-669.

[4] DENG K, ZHANG J P, YANG J. Mobile recommendation based on link community detection[J]. The Scientific World Journal, 2014: 8(26): 259156.

[5] FARUTIN V, ROBISON K, LIGHTCAP E, et al. Edge Count probabilities for the identification of local protein communities and their organization[J]. Proteins: Structure, Function, and Bioinformatics, 2006, 62(3): 800-818.

[6] NEWMAN M E J, GIRVAN M. Finding and evaluating community structure in networks[J]. Physical Review E, 2004, 69(2): 026113.

[7] BLONDEL V D, GUILLAUME J L, LAMBIOTTE R, et al. Fast unfolding of communities in large networks[J]. Journal of Statistical Mechanics: Theory and Experiment, 2008, 2008(10): P10008.

[8] TRAAG V A, WALTMAN L, VAN ECK N J. From Louvain to Leiden: guaranteeing well-connected communities[J]. Scientific Reports, 2019, 9(1): 1-12.

[9] FORTUNATO S, BARTHELEMY M. Resolution limit in community detection[J]. Proceedings of the National Academy of Sciences of the United States of America, 2007, 104(1): 36-41.

[10] RAGHAVAN U N, ALBERT R, KUMARA S. Near linear time algorithm to detect community structures in large-scale networks[J]. Physical Review E, 2007, 76(3): 036106.

[11] WANG T, CHEN S S, WANG X X, et al. Label propagation algorithm based on node importance[J]. Physica A: Statistical Mechanics and its Applications, 2020, 551(2020): 124137.

[12] LI W, HUANG C, WANG M, et al. Stepping community detection algorithm based on label propagation and similarity[J]. Physica A: statistical Mechanics and Its Applications, 2017, 472: 145-155.

[13] ROSVALL M, BERGSTROM C T. Maps of random walks on complex networks reveal community structure[J]. Proceedings of the National Academy of Sciences of the United States of America, 2008, 105(4): 1118-1123.

[14] PONS P, LATAPY M. Computing communities in large networks using random walks[J]. Journal of Graph Algorithms and Applications, 2006, 10(2): 191-218.

[15] ZHENG W P, CHE C H, QIAN Y H, et al. A two-stage community detection algorithm based on label propagation[J]. Journal of Computer Research and Development, 2018, 55(9): 1959-1971.

[16] LANCICHINETTI A, FORTUNATO S, KERTESZ J. Detecting the overlapping and hierarchical community structure in complex networks[J]. New Journal of Physics, 2009, 11(3): 033015.

[17] SHEN H W, CHENG X Q, CAI K, et al. Detect overlapping and hierarchical community structure in networks[J]. Physica A: Statistical Mechanics and its Applications, 2009, 388(8): 1706-1712.

[18] ZACHARY W W. An information flow model for conflict and fission in small groups[J]. Journal of Anthropological Research, 1977, 33(4): 452-473.

[19] LUSSEAU D, NEWMAN M E J. Identifying the role that animals play in their social networks[J]. Proceedings of the Royal Society B: Biological Sciences, 2004, 271(Suppl 6): S477-S481.

[20] GIRVAN M, NEWMAN M E J. Community structure in social and biological networks[J]. Proceedings of the National Academy of Sciences of the United States of America, 2002, 99(12): 7821-7826.

ZHENG Wen-ping, born in 1979, Ph.D, professor, Ph.D supervisor, is a member of China Computer Federation. Her main research interests include network science and bioinformatics, etc.

(责任编辑:喻藜)