doi: 10.12068/j.issn.1005-3026.2017.12.005

面向移动通信网络的局部扩张群组构造方法

李 婕[1], 王兴伟[2], 郭 静[1], 于 超[1]
(1. 东北大学 计算机科学与工程学院, 辽宁 沈阳 110169; 2. 东北大学 软件学院, 辽宁 沈阳 110169)

摘 要: 移动运营商为了拓展新业务,需要增强对用户资源的了解,因此通过大数据分析技术深入分析移动通信系统中的用户行为数据. 基于移动通信网络中的用户通话记录提出了一种基于复杂网络聚类算法的用户社交群组构造算法. 该算法通过分析用户的通话记录,建立用户间联系紧密度模型. 基于局部扩张原理和派系过滤算法进行用户群组构造. 鉴于移动通话系统的巨大数据量,采用基于 MapReduce 编程模型的并行化设计. 分别在模拟数据集和中国移动真实数据集下对该算法进行了验证,实验结果表明,该方法具有较好的性能,是可行且有效的.

关 键 词: 移动通信网络;联系紧密度;群组构造;复杂网络;MapReduce
中图分类号: TP 393 **文献标志码**: A **文章编号**: 1005-3026(2017)12-1691-06

Clique Percolation Based Local Fitness Method for User Clustering in Telecommunication Network

LI Jie[1], *WANG Xing-wei*[2], *GUO Jing*[1], *YU Chao*[1]
(1. School of Computer Science & Engineering, Northeastern University, Shenyang 110169, China; 2. School of Software, Northeastern University, Shenyang 110169, China. Corresponding author: WANG Xing-wei, E-mail: wangxw@mail.neu.edu.cn)

Abstract: To expand new business, the telecommunication companies need to understand their users deeply. So the data of the user behavior was analyzed in the telecommunication system by using the big data analyzing technology. A clique percolation based local fitness method was proposed for weighted network algorithm (CLFMw) based on the call logs of users in the telecommunication network. The social relationships were established from all of the call connections. Based on the local fitness method (LFM) and the clique percolation method (CPM), the user group was constructed with CLFMw algorithm. According to the massive data sets of the telecommunication system, parallelization design was used based on the MapReduce programming model. Finally, the group construction algorithm is verified by the simulation data set and the real data set for China Mobile. The experimental results show that this method is well performed but also feasible and effective.

Key words: telecommunication networks; relationship closeness; group construction; complex network; MapReduce

伴随着移动终端的普及,移动用户数逐渐趋于饱和,各移动运营商竞争越发激烈. 移动运营商必须深入了解用户,将更优质的服务提供给用户. 构造用户社交群组,并以此来深入了解用户,受到了移动运营商的广泛关注. 为此,本文基于移动运营商用户之间的联系紧密度,使用改进的复杂网络中的社区发现方法设计了一套用户社交群组的构造方法.

社区结构的挖掘能够帮助人们发现复杂网络中所隐藏的规律和行为. 社区发现算法分为非重

收稿日期: 2017-03-27
基金项目: 国家自然科学基金资助项目(61572123,61502092); 国家杰出青年基金资助项目(71325002); 中国博士后科学基金资助项目(2016M591449); 教育部-中国移动科研基金资助项目(MCM20160201); 中央高校基本科研业务费专项资金资助项目(N151604001).
作者简介: 李 婕(1982-),女,辽宁沈阳人,东北大学副教授; 王兴伟(1968-),男,辽宁盖州人,东北大学教授,博士生导师.

叠社区发现算法和重叠社区发现算法[1-2]. 重叠社区结构更加符合实际情况. 目前已有研究者提出了基于派系过滤、线图、局部扩张等重叠社区发现算法[3-9]. 文献[10]通过引入局部适应度函数提出了著名的 LFM(local fitness measure)算法.

LFM 算法只需要局部信息而非全网信息即可完成群组的构造,因此该算法非常适合于本文从用户数量较多的基于社交关系的复杂网络中挖掘用户的社交关系群组. 然而,LFM 主要针对无权网络,并不能直接用于本文所抽象出的有权复杂网络. 而且,LFM 算法初始时选择节点具有一定的随机性. 为了能使该算法有效地适应有权复杂网络中,降低算法的随机性,保证算法运行的质量,本文对 LFM 算法进行了改进,由于改进后的 LFM 算法结合了 CPM(clique percolation method)算法[11]思想并且可用于处理有权网络,所以将改进后的算法命名为 CLFMw(clique percolation based local fitness method for weighted network).

1 问题描述

1.1 复杂网络模型

将移动通信网络中的用户抽象为节点,用户间的联系紧密度值抽象为边的权值,将用户间的通信关系抽象为有权复杂网络.

1.2 群组构造

基于所抽象出的有权复杂网络,改进复杂网络中的优秀社区发现方法作为社交群组构造算法,使其能够应用于本文的有权复杂网络并能适当提高其群组构造的质量.

2 算法设计

LFM 算法是基于局部扩张原理的经典社区发现算法之一. LFM 算法将群组定义为拥有最大局部适应度值的子图,其适应度函数定义为

$$f_c = \frac{k_{in}^c}{(k_{in}^c + k_{out}^c)^{\alpha}}. \tag{1}$$

其中:k_{in}^c表示群组 c 内部节点的边数之和;k_{out}^c表示群组 c 的内部节点和外部节点的边数之和;α是一个正实数,用于控制社区的大小.

LFM 算法首先在网络中随机选择一个节点作为种子节点并以该种子节点为初始群组,然后不断向其邻接节点扩张,直至群组的适应度函数 f_c 达到局部最优. 随后,算法再次从网络中随机选择一个未被划分至任何群组的节点作为种子节点,迭代执行上述过程,直至所有节点均划分至一个或者多个群组为止. 由于各个群组在局部扩张过程中完全独立,相互间无任何影响,因而某一个节点可能同时被划入多个社区,所以该算法能够发现重叠社区.

然而,尽管 LFM 能够发现重叠社区,但该算法主要针对无权网络,并不能直接用于本文所抽象出的有权复杂网络,所以本文将 LFM 算法的适应度函数进行了改写,使其适用于有权网络. 此外,LFM 算法初始时随机选择节点进行群组扩张,这致使该算法具有一定的随机性,且随机选择的节点质量也直接决定着群组构造的结果,并且在实际社交关系网络中,很难定位较好种子节点,即便发现了较好的种子节点,只单一通过一个种子节点作为初始群组进行群组构造也不能得到较好的群组构造效果. 鉴于此,CLFMw 算法使用种子群组代替种子节点作为算法运行的初始群组. 除此之外,CLFMw 算法还判断节点是否应该加入群组时的联系度约束条件,并且为防止群组重叠率过高而进行了群组合并.

2.1 联系紧密度计算

用户 A 对用户 B 的联系强度计算方法如式(2)所示.

$$S_{AB} = \left(\frac{AVG_{couple\_duration}}{AVG_{all\_duration}} \times \frac{FRE_{couple\_times}}{AVG_{all\_times}}\right)^{\frac{1}{2}}. \tag{2}$$

其中:$AVG_{couple_duration}$表示 A 与 B 的平均通话时长;FRE_{couple_times}表示 A 与 B 的总通话次数;$AVG_{all_duration}$表示 A 与其所有通话对象的平均通话时长的均值;AVG_{all_times}表示 A 与其所有通话对象的平均通话次数.

联系稳定性的度量如式(3)所示.

$$C_{AB} = \left(\frac{FRE_{couple\_weeks\_times}}{AVG_{all\_weeks\_times}} \times \frac{CV_{gap\_weeks}}{AVG\_CV_{gap\_weeks}}\right)^{\frac{1}{2}}. \tag{3}$$

其中:$FRE_{couple_weeks_times}$表示 A 与 B 的总通话周数;$AVG_{all_weeks_times}$表示 A 与其所有通话对象的平均通话周数;CV_{gap_weeks}表示 A 与 B 联系间隔周数的离散系数;$AVG_CV_{gap_weeks}$表示 A 与其所有通话对象的联系间隔周的离散系数均值.

以 A 为主体定义 A 对 B 的联系紧密度的计算方法如式(4)所示.

$$I_{AB} = \theta C_{AB} + (1-\theta) S_{AB}. \tag{4}$$

其中,$\theta \in [0,1]$,用于调节联系强度与联系稳定性对联系紧密度的影响程度的常量.

从用户 B 的角度也可以计算出 B 对 A 的联系紧密度 I_{BA},因此定义 A 与 B 的综合联系紧密度如式(5)所示.

$$I=\frac{n_{AB}}{n}I_{AB}+\frac{n_{BA}}{n}I_{BA}. \tag{5}$$

其中：n_{AB}表示 A 主叫 B 的通话次数；n_{BA}表示 B 主叫 A 的通话次数；显然 $n=n_{AB}+n_{BA}$.

2.2　种子群组构造

本文在进行群组构造前先构造一个种子群组，种子群组内的节点应为群组内联系非常紧密的用户集合，它们可以作为群组的核心. 随后以该种子群组为其他群组构造算法的初始群组，其他群组构造算法在该种子群组的基础上进行群组构造. 派系过滤算法是通过全连通子图来构造群组，而全连通子图是网络中节点间联系最为紧密的节点集合，因此其所构造的群组具有较强的群组特性. 鉴于此，本文采用基于派系过滤的 CPMw 算法构造种子群组. 为了使构造的种子群组具有层次性，从最大的派系直至最小的派系（2 – 派系）逐级过滤构造种子群组，算法流程如下.

算法 1　基于非固定派系大小的种子群组构造算法

输入：有权复杂网络

输出：种子群组

BEGIN

1：Initialize var $k=1$

2：**DO**

3：　$k++$

4：　Calculate satisfied k-clique

5：**WHILE** k-clique is not the maximum clique

6：**WHILE** $k\geqslant 2$

7：　Delete k-cliques that have belonged to other groups

8：　Percolate k-clique

9：　Construct seed groups based on k-clique

10：　$k--$

11：**END WHILE**

END

2.3　适应度函数

群组的适应度函数主要用来衡量当前正在构造群组的群组特性，适应度函数越大说明当前群组的群组特性越强. 本文将适应度函数定义为

$$\mathrm{Fit}_{\mathrm{cluster}}(C)=\frac{B_{\mathrm{in}}}{(B_{\mathrm{in}}+B_{\mathrm{out}})^{\alpha}}=\frac{\sum_{i,j\in C}w_{i,j}}{(\sum_{i\in C}w_{i,j})^{\alpha}}. \tag{6}$$

其中：C 表示当前正在构造的群组；B_{in}表示群组内部节点间的边的权值之和；B_{out}表示群组内部节点与群组外部节点连接边的权值之和；w_{ij}表示节点 i 与节点 j 连接边的权值；若节点 i 与节点 j 间不存在边则 $w_{ij}=0$；α 为正数常量，可用于调节社区的规模.

群组的适应度函数随着邻接节点的加入而不断变化，节点 i 加入群组后，群组的适应度函数如式（7）所示.

$$\mathrm{Fit}^{i}_{\mathrm{cluster}}(C)=\frac{B_{\mathrm{in}}+b^{i}_{\mathrm{in}}}{(B_{\mathrm{in}}+b^{i}_{\mathrm{in}}+B_{\mathrm{out}}-b^{i}_{\mathrm{in}}+b^{i}_{\mathrm{out}})^{\alpha}}=\frac{B_{\mathrm{in}}+b^{i}_{\mathrm{in}}}{(B_{\mathrm{in}}+B_{\mathrm{out}}+b^{i}_{\mathrm{out}})^{\alpha}}. \tag{7}$$

其中：b^{i}_{in}表示节点 i 与群组 C 内的节点间边的权值之和；b^{i}_{out}表示节点 i 与群组 C 之外节点间的边的权值之和.

将上述公式进行变形，可得式（8）.

$$\mathrm{Fit}^{i}_{\mathrm{cluster}}(C)=\frac{B_{\mathrm{in}}}{(B_{\mathrm{in}}+B_{\mathrm{out}})^{\alpha}}\times\frac{1+\frac{b^{i}_{\mathrm{in}}}{B_{\mathrm{in}}}}{(1+\frac{b^{i}_{\mathrm{out}}}{B_{\mathrm{in}}+B_{\mathrm{out}}})^{\alpha}}. \tag{8}$$

定义节点 i 的适应度函数为节点 i 加入群组后的群组模块适应度与节点 i 加入群组前的群组模块适应度差值，即如式（9）所示

$$\mathrm{Fit}^{i}_{\mathrm{node}}=\Delta\mathrm{Fit}_{\mathrm{c}}(C)=\mathrm{Fit}^{i}_{\mathrm{c}}(C)-\mathrm{Fit}_{\mathrm{c}}(C). \tag{9}$$

CLFMw 算法将通过计算节点适应度函数 $\mathrm{Fit}^{i}_{\mathrm{node}}$的大小是否大于增量阈值 $\mathrm{Fit}^{i}_{\mathrm{node}}$决定该节点是否加入当前正在构造的群组.

2.4　联系度约束

联系度约束是为了避免如图 1 所示的情况发生，比如用户 B 只是用户 A 的一个私人朋友而不应属于群组 C. 组外的邻接节点除了需要满足该节点的适应度函数大于阈值 $\mathrm{Fit}^{i}_{\mathrm{node}}$外，还应满足该节点与群组内的节点的联系边数 N_{in}不小于当前群组内节点总数 N 的 z %，其中 z 为一阈值.

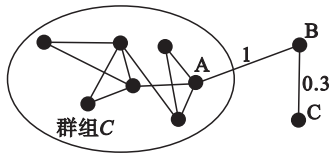

图 1　错误的群组构造情况

Fig. 1　Incorrect group construction

2.5　群组合并

定义群组 C_1 与群组 C_2 重叠率 InterRate（C_1，C_2）如式（10）所示.

$$\mathrm{InterRate}(C_1,C_2)=\frac{|C_1\cap C_2|}{\min(|C_1|,|C_2|)}. \tag{10}$$

其中：$|C_1\cap C_2|$表示群组 C_1 与群组 C_2 交集的节点个数；$\min(|C_1|,|C_2|)$表示群组 C_1 与群组 C_2 所含节点个数的最小值. 当重叠率较大时，重叠的

群组极为可能为同一个群组而被重复构造,应该予以合并. 设定重叠率阈值 InterRate*,如果两个群组间的重叠率 InterRate(C_1, C_2)大于重叠率阈值 InterRate*,则将此两个群组进行合并.

2.6 CLFMw 算法流程

综上所述,CLFMw 算法的流程图如下. 该算法采用种子节点逐级注入的形式,较大的群组可以通过基于高派系的种子群组进行构造,较小的群组可以通过基于低派系的种子群组进行构造,所以该算法同样能够覆盖网络中所有节点.

算法 2 CLFMw 群组构造算法

输入:有权复杂网络
输出:群组结构
BEGIN
Construct seed groups
WHILE existing seed groups not belonging to any groups
  Initialize a seed group which based on largest cliques as initial group
  WHILE current group exists adjacency nodes
    FOR each adjacency node
      Calculate $\mathrm{Fit}_{\mathrm{node}}, N_{\mathrm{in}}$
      IF $\mathrm{Fit}_{\mathrm{node}} \geqslant \mathrm{Fit}^*_{\mathrm{node}}$ & $N_{\mathrm{in}} \geqslant N \times z\%$
        Add this node into group
      END IF
    END FOR
    Calculate InterRate(C_1, C_2)
    IF InterRate$(C_1, C_2) \geqslant$ InterRate*
      Merge group C_1 and group C_2
    END IF
  END WHILE
END WHILE
END

3 性能评价

3.1 实验平台

本文对所设计的 CLFMw 群组构造方法在 Hadoop 平台下进行了基于 MapReduce 的并行实现,并使用 LFR 基准网络、中国移动真实通话记录数据集分别进行了性能评价.

3.2 并行化设计

CLFMw 群组构造算法基于局部扩张原理,其基于不同种子群组的各个群组构造部分相互独立、无任何影响,因此可以将各个群组构造过程并行化. 为了使 CLFMw 算法既能高效并行又能同时尽量避免出现大范围的高度重叠的群组,算法采用逐级注入所有基于某一派系的种子群组注入方式. 首先将所有基于最大派系生成的种子群组作为 CLFMw 算法的种子群组并进行并行的群组构造,当群组构造结束,再次将所有基于次大派系的种子群组作为 CLFMw 算法的种子群组进行并行的群组构造,直至基于 2 - 派系的种子群组注入并群组构造完毕.

3.3 LFR 基准网络实验结果分析

1)实验背景. 采用基准网络 LFR[11]. 所选取的对比算法为基于标签传播原理的 COPRA 算法[7]和基于局部扩张原理的 OSLOM 算法[10],这两种算法均具有较好的性能[3]. 采用扩展标准互信息(extended normal mutual information, ENMI)[10]作为性能对比的指标. LFR 网络参数参照文献[3]配置,通过分别调整拓扑混合参数 μ_{t}、权值混合参数 μ_{w}、重叠节点的个数 O_{n} 来深入观察各个群组构造算法的性能.

2)拓扑混合参数 μ_{t} 对算法的性能影响. 拓扑混合参数 μ_{t} 指节点外部度数占其总度数的比例,设置 LFR 基准网络中 $N = 50\ 000$, $\mu_{\mathrm{w}} = 0.1$, $O_{\mathrm{n}} = 5\ 000$,调整 μ_{t},对比结果如图 2 所示,CLFMw 群组构造算法的性能明显好于对比算法,这主要是因为随着 μ_{t} 不断增大,网络中的群组拓扑开始变得不清晰,CLFMw 算法所构造的种子群组具有非常强的群组特性,因此在此情况下性能良好.

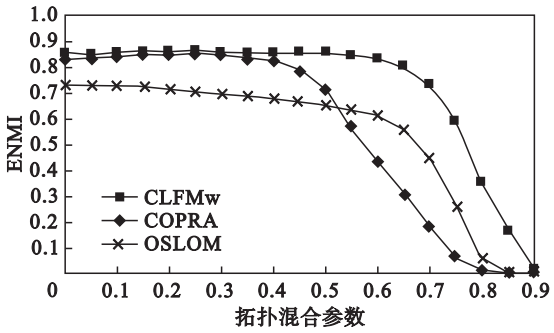

图 2 拓扑混合参数对构造群组算法的影响

Fig. 2 Influence of topological mixing parameters on the group algorithm

3)权值混合参数 μ_{w} 对算法性能的影响. 权重混合参数 μ_{w} 是节点对群组外节点连接边的权值总和与该节点与所有节点连接边的权值总和的比例值,设置 LFR 基准网络中 $N = 50\ 000$, $\mu_{\mathrm{t}} = 0.2$, $O_{\mathrm{n}} = 5\ 000$,调整 μ_{w},结果如图 3 所示,随着 μ_{w} 的增大,算法的性能开始出现明显的差异,这主要是因为种子群组在较模糊的群组结构中识别

了群组内的核心群组关系，致使其性能好于 COPRA 算法.

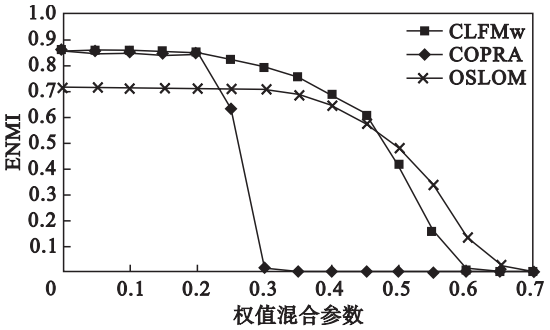

图 3　权值混合参数对构造群组算法的影响

Fig. 3　Influence of the mixed parameters of weights on the group algorithm

4）重叠节点数 O_n 对算法性能的影响. 重叠节点数 O_n 是指基准网络中重叠节点的个数. 设置 LFR 基准网络中 $N = 50\ 000$，$\mu_t = 0.3$，$\mu_w = 0.2$，调整 O_n，结果如图 4 所示，CLFMw 算法的性能明显好于 OSLOM 和 COPRA 算法. 这是因为算法初始时所注入的种子群组即有重叠，有利于算法发现重叠群组. 而 COPRA 和 OSLOM 算法均相当于从一个节点作为初始群组进行群组构造. 此外，CLFMw 群组构造算法中各个群组构造过程完全独立也是高质量地构造重叠群组的保证.

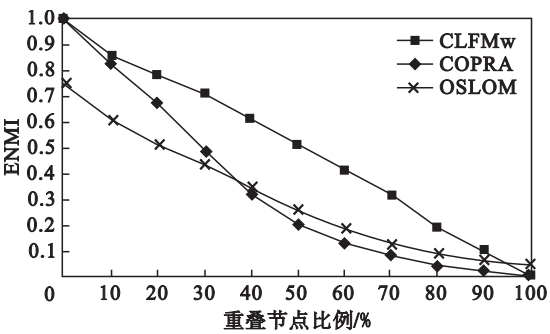

图 4　重叠节点数对构造群组算法的影响

Fig. 4　Influence of the number of overlapping nodes on the group algorithm

3.4　移动通信数据集分析

1）实验背景. 实验验证过程中，共度量出 4 406 891位用户，33 728 562 条有权关系，平均每位用户拥有 7.654 条关系. 图 5 为联系紧密度值所对应关系数的分布图，从分布角度而言，联系紧密度值基本符合幂律分布，符合实际情况.

2）群组构造质量评估. 因为用户的真实群组划分是未知的，所以无法使用标准互信息作为衡量指标，可以采用聚集系数来衡量群组构造的质量. 结果如表 1 所示，可以看出，各个群组内节点的聚集系数、加权聚集系数的均值都远大于全网所有节点的聚集系数、加权聚集系数，其比值均在 4.5 倍以上，CLFMw 群组构造算法的群组构造质量均较高，算法是可行且有效的.

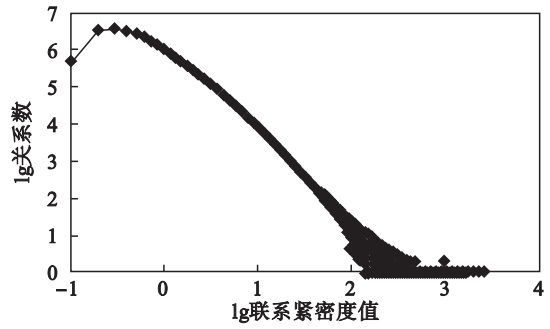

图 5　用户联系紧密度值的关系数分布图

Fig. 5　Distribution of the number of user relationship tightness

表 1　群组构造算法中聚集系数比较图

Table 1　Comparison of clustering coefficients in group construction algorithm

群组构造算法	聚集系数均值			加权聚集系数均值		
	群组	全网	比值	群组	全网	比值
CLFMw	0.663	0.143	4.636	0.483	0.095	5.084

4　结　　论

本文基于移动用户通话记录设计了一套社交群组构造方法，首先基于通话记录度量了用户间的联系紧密度，然后基于此联系紧密度构建复杂网络，基于局部扩张原理和派系过滤算法设计了群组发现算法，使其能够针对有权复杂网络进行群组构造并保证群组构造的质量. 鉴于移动通信网络中的数据量巨大，本文对所改进设计的群组构造算法进行了并行化实现，并对其性能进行了验证. 实验结果表明，所设计的群组构造算法具有较好的性能，是可行且有效的.

参考文献：

[1] 刘大有，金弟，何东晓，等. 复杂网络社区挖掘综述[J]. 计算机研究与发展，2013，50(10)：2140－2154.
(Liu Da-you，Jin Di，He Dong-xiao，et al. Community mining in complex networks[J]. *Journal of Computer Research and Development*，2013，50(10)：2140－2154.)

[2] Harenberg S，Bello G，Gjeltema L，et al. Community detection in large-scale networks：a survey and empirical evaluation [J]. *Wiley Interdisciplinary Reviews：Computational Statistics*，2014，6(6)：426－439.

[3] Xie J R，Kelley S，Szymanski B K. Overlapping community detection in networks：the state-of-the-art and comparative study[J]. *ACM Computing Surveys*，2013，45(4)：43.

（下转第 1711 页）