软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2019,30(4):1045−1061 [doi: 10.13328/j.cnki.jos.005387]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于节点向量表达的复杂网络社团划分算法*

韩忠明[1,2], 刘 雯[1], 李梦琪[1], 郑晨烨[1], 谭旭升[1], 段大高[1]

[1](北京工商大学 计算机与信息工程学院,北京 100048)

[2](食品安全大数据技术北京市重点实验室,北京 100048)

通讯作者: 韩忠明, E-mail: hanzm@th.btbu.edu.cn

摘 要: 社团结构划分对复杂网络研究在理论和实践上都非常重要.借鉴分布式词向量理论,提出一种基于节点向量表达的复杂网络社团划分方法(CDNEV).为了构建网络节点的分布式向量,提出启发式随机游走模型.利用节点启发式随机游走得到的节点序列作为上下文,采用 SkipGram 模型学习节点的分布式向量.选择局部度中心节点作为 *K*-Means 算法的聚类中心点,然后用 *K*-Means 算法进行聚类,最终得到社团结构.在真实和模拟两种网络上做了丰富的实验,与主流的全局社团划分算法和局部社团划分算法作了比较.在真实网络上 CDNEV 算法的 *F*1 指标比其他算法平均提高 19%;在模拟网络上,*F*1 指标则可以提高 15%.实验结果表明,相对其他算法,CDNEV 算法的精度和效率都较高.

关键词: 复杂网络;社团结构;核心节点;结构关系强度

中图法分类号: TP311

中文引用格式: 韩忠明,刘雯,李梦琪,郑晨烨,谭旭升,段大高.基于节点向量表达的复杂网络社团划分算法.软件学报,2019,30(4):1045−1061. http://www.jos.org.cn/1000-9825/5387.htm

英文引用格式: Han ZM, Liu W, Li MQ, Zheng CY, Tan XS, Duan DG. Community detection algorithm based on node embedding vector representation. Ruan Jian Xue Bao/Journal of Software, 2019,30(4):1045−1061 (in Chinese). http://www.jos.org.cn/1000-9825/5387.htm

Community Detection Algorithm Based on Node Embedding Vector Representation

HAN Zhong-Ming[1,2], LIU Wen[1], LI Meng-Qi[1], ZHENG Chen-Ye[1], TAN Xu-Sheng[1], DUAN Da-Gao[1]

[1](School of Computer and Information Engineering, Beijing Technology and Business University, Beijing 100048, China)

[2](Beijing Key Laboratory of Big Data Technology for Food Safety, Beijing 100048, China)

Abstract: Community detection is very important in theoretical and practical for complex research. According to the principle of distributed word vector, a community detection algorithm based on node embedding vector (CDNEV) is proposed in this study. In order to construct the distributed vector of network nodes, a heuristic random walk model is put forward. The node sequence obtained by the heuristic random walk model is used as the context for nodes, and the distributed vector of nodes is learned by SkipGram model. Based on the distributed vector of nodes that are selected from the local node as the center of the *K*-Means clustering algorithm center, all nodes in a network are clustered with *K*-Means algorithm, and the community structure are conclude by clustering result. Based on real complex networks and artificial networks used in other state-of-the-art algorithms, comprehensive experiments are conducted. For comparison purpose, typical community detection algorithms are selected to be evaluated. On real networks, the *F*1 value of CDNEV algorithm is increased 19% on average. The *F*1 value can be increased by 15% on artificial networks. Experimental results demonstrate that both accuracy and efficiency of CDNEV algorithm outperform other state-of-the-art algorithms.

Key words: complex network; community detection; key node; structural strength

* 基金项目: 国家自然科学基金(61170112, 61532006); 北京市自然科学基金(4172016, KZ201410011014)

Foundation item: National Natural Science Foundation of China (61170112, 61532006); Natural Science Foundation of Beijing, China (4172016, KZ201410011014)

收稿时间: 2016-10-09; 修改时间: 2017-06-09, 2017-08-25; 采用时间: 2017-09-09

1 引 言

如何准确、高效地进行社团结构划分是当前复杂网络理论研究领域中的热点问题.很多复杂的系统可以转化为由节点和边组成的网络来进行建模,如人与人之间的关系网络、科学家之间的合作网络、学术论文间相互引用网络、蛋白质交互网络和 WWW 链接网络等.网络中一个十分重要的性质是社团结构,理解社团结构对理解网络的整体具有重要价值.Newman 把社团结构定义为网络中的若干个群体或者团体,群体内部的节点间存在紧密连接,群体间的连接则相对稀疏[1].

复杂网络社团划分对人们了解真实的网络信息具有非常重要的意义.社团是由复杂网络中具有相同性质的个体组成,例如万维网中的社团可以是提供相似内容的网站,蛋白质网络中的社团可以是具有相似功能的基因.发现复杂网络中的社团结构可以帮助人们了解复杂系统的拓扑性质和组织结构.很多研究表明,复杂网络的社团结构特征不同于网络的全局结构特征[2].因此,复杂网络社团结构划分一直是重要的研究领域.近年来,很多研究者从网络拓扑结构、节点属性等各种特性上研究社团划分的方法.由于复杂网络具有结构稀疏、节点度分布呈现幂率分布和长尾分布等特性,基于拓扑结构的社团划分算法在实际复杂网络中的划分效果并不理想.聚类技术也可以应用于社团划分,聚类可以利用节点的结构特性、描述属性等,使用聚类结果作为社团划分的依据.聚类方法一般需要对节点表示成一个稀疏的高维向量,高维稀疏向量不仅难以有效地表达结构的上下文结构信息,而且高维聚类算法复杂度也较高.

深度学习被广泛应用于机器学习、自然语言处理中.Word2Vec 巧妙地利用了词的上下文结构,采用三层神经网络对词构造分布式向量表达.自从词向量提出后,很多研究者对不同的研究对象进行了向量化.用低维的实向量表达一个对象具有维度低、包含上下文信息等优点,所以得到了很多研究者的关注.文献[3]分析了复杂网络中随机游走产生的节点序列的频率,发现其和文本分析中的词一样具有长尾效应,满足 zipf 定律,首次提出一种网络节点的向量表示方法,并在标签预测上进行了应用实验,实验结果表明,节点向量能够表示节点所在的结构特性,所以在标签预测上效果较好.

受其启发,本文将分布式节点向量表达和聚类方法相结合应用于复杂网络的社团划分[4,5],在利用分布式节点向量进行社团划分时,期望节点的上下文能够较好地刻画社团内部和外部的差异,所以本文提出一个启发式随机游走模型,将启发式随机游走的路径作为节点上下文,采用分层回归(hierarchical softmax)方法生成节点向量,利用节点向量进行快速聚类,实现社团划分.

2 研究现状

近年来,复杂网络社团划分研究受到国内外研究者的大量关注,他们提出了不同类型的复杂网络社团划分方法,主要可以分为以下 3 类.

(1) 基于优化的划分方法,其代表是谱方法和聚类方法.谱方法[6,7]可以用来进行社团划分,其原理是由网络邻接矩阵的特征值和其对应特征向量来划分网络的社团结构.付立东等人[8]基于核矩阵的最大特征向量和特征值提出了模块密度中心性,用来度量节点对不同社团的贡献度,在此基础上划分社团.Riolo 等人[9]为了能够使网络的最小分割矩阵映射到谱中,对网络的最小分割矩阵进行了优化.然而,谱方法在网络中社团的数量识别方面存在着明显的缺陷,因为通过递归二分策略得到的划分结果不一定是最优解.Liu 等人[10]将节点的中心度和节点间的吸引力分别作为节点和连边的权重,在聚类时保证每个簇内节点的权重之和与簇和簇之间节点连边权重之和的差值最大,在此基础上提出了基于节点中心性和节点间吸引力的加权节点聚类算法.Wang 等人[11]利用网络的潜在拓扑结构的拉普拉斯矩阵提出了基于网络潜在拓扑结构的谱聚类算法,该算法充分利用网络的拓扑结构信息,并且利用局部最大潜在节点对划分的最终社团数目进行优化.Jin 等人[12]利用物理拓扑距离度量聚类时簇的距离,在此基础上提出了基于密度的社团划分方法.Lin 等人[13]发现在网络的结构信息不完整时,可以通过已知局部区域的边信息来度量网络(缺失部分边的条件下)中节点间的距离,并进一步利用层次聚类划分网络的社团结构.Gennip 等人[14]将谱聚类算法应用于地理定位数据中的社团发现,并对不同地理信息编码方

式对最终划分结果的影响进行了探索.Kernighan 等人[15]基于贪婪思想提出了一种试探优化社团划分算法.Newman 快速算法[16]通过不停地在网络中添加边来快速、有效地对大型网络进行社团划分.Ying 等人[17]提出通过迭代的方式把每个节点划分到与其拥有最大相似度的社团来发现复杂网络中的社团结构.Zanjani 等人[18]从节点间的连接关系的紧密程度和依赖关系的强弱程度来判断节点间的关系,将关系紧密的节点划分到同一个社团中,进一步对复杂网络进行社团划分.王兴元等人[19]利用节点间依赖度进行社团划分.Clauset 等人[20]基于 R 值提出了局部社团发现算法,R 为局部模块度,然后基于贪心算法的思想迭代地添加邻居节点,在此过程中,要求 R 值增加最大化.Lancichinetti 等人[21]基于社团局部结构定义一个适应度函数,在此基础上提出了基于适应度函数的重叠社团划分算法(LFM).针对 LFM 算法随机选择初始种子节点,可能存在算法陷入死循环的问题,Lee 等人[22]选择极大子图 clique 作为初始社团取代 LFM 算法中种子节点随机选择策略,在此基础上提出 GCE 算法.此类算法需要社团数量或社团的平均规模大小等先验知识,并且对初值的选取比较敏感,若初值的选取策略不当可能会导致算法不易收敛,结果较差;Newman 快速算法等基于贪心算法思想的方法,在推广至大规模社团划分场景时,存在收敛速度慢、时间复杂度较高等问题.Xu 等人[23]将网络中的节点划分为社区节点、枢纽节点和边缘节点这 3 类节点,提出了 SCAN 算法用于对 3 类节点的聚类分析.Huang 等人[24]提出了一种 SHRINK 模型,该模型在枢纽节点及边缘节点的基础上,加入了社团的层级结构信息,在真实数据集上取得了很多好的划分结果.

(2) 启发式划分方法,主要是基于网络结构的社团划分算法.GN(Girvan-Newman)算法[25]的提出对社团划分研究的发展有着重要意义,Girvan 和 Newman 创新性地提出了“边介数”,边介数较高的边更有可能是社团之间的连边,删除这些边介数高的边可以很好地划分社团.GN 算法的精确度有了显著的提高,然而其时间复杂度却非常高.Vincent 等人[26]基于模块度优化提出了一种启发式社团划分算法 Fast Unfoliding.算法包括两个步骤:首先将每个节点分配到指定的社团,然后在社团间按序移动分配好的节点;然后将上一阶段得到的社团作为新的“节点”,重新构造新的子图.Raghavan 等人[27]基于标签传播算法(LPA)进行社团划分,LPA 为每个节点指定一个标签,然后迭代更新节点的标签,直到所有节点达到收敛为止.Sun 等人[28]提出了一种基于中心的标签传播算法 CenLP,该方法主要通过计算本地密度和节点与高密度节点的相似性来改进标签传播算法,该方法基本不需要人工设置参数,就可以适用于较大规模的网络数据.Tsourakakis 等人[29]在两种正交的启发式搜索算法的基础上提出一个框架,其中,这两种启发式算法可以通过权值来进行量化,这个框架能够应用到传统的社团划分算法中,使它们的性能得到提高.袁超等人[30]提出了节点相似度模型应用于局部社团挖掘,进一步使用“内外夹逼”的思想,提出了一种有效的社团发现算法(SICE).这类算法的共同特点是它们都是基于某些直观的假设来设计启发式算法,对于大部分网络来说,它们能够快速寻得最优解或近似解,但无法从理论上保证算法对具有任何结构的输入网络都能找到最优或近似最优解.

(3) 其他划分方法.Albert 等人[31]利用双曲率对网络结构进行有限的描述,可以很好地发现网络中的一些具有高阶连通性的节点,并将这些节点作为社团发现的重要节点.淦文燕等人[32]基于拓扑势提出了一种新的社团划分算法,算法将拓扑势场的局部高势区看成社团,并且连通的高势区被低势区所分割,利用网络的这些高势区可以发现网络中的社团.Wu 等人[33]将复杂网络类比为电路系统,网络中的边类比为电阻,边连接的节点之间存在电势差,在此基础上提出了快速启发式社团发现算法(WH).何东晓等人[34]基于聚类融合提出了一种遗传算法,并将其应用于社区发现,遗传算法利用父节点的聚类信息结合网络的局部拓扑结构信息,产生新节点.Okamoto 等人[35]提出将复杂网络中的社团结构视为神经网络中的细胞集合进行社团划分.张忠元等人[36]在网络社团发现问题中引入字典学习算法形成新的算法,该算法结合了字典学习算法和最小二乘方回归,算法更加简单,收敛速度更快,且在社团发现中取得了较好的效果.除了将物理、生物等自然科学学科知识引入复杂网络社团划分外,有研究者尝试将复杂网络中节点或边进行向量化表示,利用得到的节点或边向量进行社团划分.Tang 等人[37]基于社交网络中人与人之间的交互行为,提出将节点作为边的特征对边进行向量化表示,这种边缘-中心算法能够处理稀疏网络并具有可扩展性等特点.

基于优化的划分方法和启发式划分方法存在各自的缺陷,如何利用节点的结构特性,构造具有良好性能的

社团划分算法是本文研究的目标.本文将自然语言处理(NLP)中的语言模型应用于复杂网络节点特征向量的学习,节点的特征向量在一定程度上可以反映复杂网络中的信息,最后通过聚类的方法完成网络中节点所属社团的识别.近年来,随着深度学习的发展,也产生了很多新的节点表示学习方法,如基于随机游走的模型 DeepWalk、node2vec,基于近邻相似性的 LINE[38]、GraRep[39]、SDNE[40],引入其他属性、文本信息[41,42]的 TADW[43]、GENE[44]等,这些方法虽然不同程度地包含了网络的结构、属性、文本等信息,但目前针对社团划分任务的网络表示方法仍有待研究.

3 基于节点向量表达的复杂网络社团划分算法

本文提出一种基于节点向量表达的复杂网络社团划分算法(community detection algorithm based on node embedding vector,简称 CDNEV).

CDNEV 算法主要包含 3 个步骤,为了对节点生成分布式向量表达,首先需要构造包含节点上下文的语料库.节点上下文语料库对社团划分具有重要作用,为了克服随机游走算法带来的节点上下文不能很好地刻画社团结构的特性,我们提出一种启发式随机游走算法,对每个节点生成固定长度的上下文节点序列,作为语料库;然后利用 SkipGram 模型来生成分布式节点表示向量;最后利用特征向量进行聚类,得到复杂网络中社团划分的结果.下文将使用的数学符号列于表 1 中.

Table 1 Main symbols

表 1 主要符号列表

n	m	M	k	N	L	X
网络节点数	随机游走数量	迭代网络次数	随机游走步长	字典大小	句子长度	社团数目
d	w	r	P	v		
节点向量维数	序列窗口	遍历网络次数	网络节点度平均度	网络边数量		

3.1 启发式随机游走

在语言模型中,词是处理的基本对象,句子是词的上下文载体,句子集构成语料库.在复杂网络上,节点是处理的基本对象,但没有承载节点的句子,文献[3]中采用了随机游走生成节点的上下文节点序列,这样构造出由节点序列形成的“句子”,在节点上进行不同的随机游走可以得到不同的句子,形成语料库.

随机游走具有不确定性.一般地,自然语言中的句子是满足特定语法规则和含义的词串,若采用完全随机游走的方式把复杂网络的节点序列转化为句子,容易产生许多随机性很强的句子,这些句子相当于自然语言中的不符合语法或噪声很大的句子,对语言模型训练出来的词向量(节点的特征向量)会带来负面效果.因此,我们提出一种启发式随机游走算法,首先通过随机游走生成加权的 k-step 图,然后在 k-step 图上进行概率随机游走,提高生成节点序列的社团合理性.

3.1.1 启发式随机游走相关定义

在语言模型中,随机游走过程主要表示为:网络上的任意一个节点依照某一概率,从当前位置转移到与其有边连接的邻居节点的过程.令 G 表示一个给定节点数为 n 的加权复杂网络,其邻接矩阵为 $A=|a_{ij}|_{n\times n}$,a_{ij},表示节点 i 和 j 的连接布尔值.边权 $w(e_{ij})$表示节点 i 和 j 连边的权重,将其初始化为 1.

复杂网络中相距较远的两个节点之间的交互行为一般较少,相互影响较弱.启发式随机游走算法中相关定义如下.

定义 1(k-step). k-step 表示在图 G 中进行随机游走,当步长为 k 或遇到预设停止条件时,结束当次随机游走,其中,$0<k\leqslant N$.

定义 2(k-step 概率). 令 $e\in E$ 表示图 G 的一条边,k-step 概率 $P_k(e)$表示从当前节点进行随机游走过程选择通过边 e 的概率.

定义 3(预处理 k-step 图). 预处理 k-step 图是图 G 中每个节点通过 m 次步长为 k 的完全随机游走生成的加权图,其中,完全随机游走指的是当前节点所对应的每一条边的 k-step 概率的值相同,边的权重为该边在完全

随机游走过程中被遍历的次数.

生成预处理 k-step 图的基本过程如下:令当前节点所对应的每一条边的 k-step 概率为相同值,即以等概率选择任意一条边作为下一步.每经过一条边就将该点对应的 $w(e_{ij})$加 1,循环 m 次游走过程.

因为复杂网络社区具有“内紧外松”的特性,所以,社区内节点间的交互行为比社区之间的联系更为频繁.预处理 k-step 图中的完全随机游走过程中经过社团内部的次数应该明显多于经过社团间的次数.由此可得:经过预处理的 k-step 图中每一个节点与其邻节点所形成的连边中,边的权值越大,其对应的 k-step 概率越大,最终在生成节点序列的启发随机游走的过程中,从当前节点经过该边的概率越大,从而形成对社区划分更合理的“句子”.

3.1.2 启发式随机游走算法

算法 1. RandomWalkByWeight (G,v_i,k).

输入:Graph $G(V,E)$; start vertex v_i; step size k;

输出:Node sequence W_{v_t}.

1: Initialization: Calculate the probability of e is selected, $e\in E$

2: for i=0 to k do

3: select the next step (e) in probability

4: add in W_{v_t}

5: end for

6: if the W_{v_t} of different nodes is less than $\frac{2}{3}k$ then

7: drop W_{v_t}

8: end if

根据第 3.1.1 节所述,可以将算法分为生成预处理 k-step 图和 k-step 启发式随机游走两个部分.生成预处理 k-step 图的流程如下.

Step 1: 把复杂网络的每一条边的权值 $w(e_{ij})$初始化为 1.

Step 2: 将图中的每一个节点依次作为源节点进行完全随机游走.完全随机游走即从当前节点出发,对于所有邻边将其作为一条路径的概率均相等.完全随机游走过程有 k 步,每经过一条边就将该点对应的 $w(e_{ij})$加 1.

Step 3: 重复 m 次 Step 2,生成经过预处理的 k-step 图,即得到该网络预处理后的最终边权,易知,某一条边的权值越大,这条边对识别社团的贡献越大.

综上所述,假设复杂网络中节点数为 n,随机游走长度为 k,迭代网络次数为 M,那么生成预处理 k-step 图的时间复杂度为 $O(Mkn)$.

k-step 启发式随机游走是在图 G 上利用预处理 k-step 图生成的权重进行加权随机游走,具体算法如算法 1 所示.算法包含 4 个主要步骤.

Step 1: 对生成的预处理 k-step 图中每个节点与相连的边进行归一化处理,把与一个节点相连的所有边的权值转化为相应的概率(line 1,算法 1).设节点 i 与节点 j 之间的连边被选中的概率为

$$P(e_{ij})=\frac{w(e_{ij})}{\sum_{e\in E(v_i)}w(e)},$$

易知 $\sum_{e\in E(v_i)}P(e)=1$, 其中,$E(v_i)$为节点 i 与其邻接点所形成的连边.

Step 2: 选取指定节点作为源节点进行 k-step 随机游走,k-step 过程根据 Step 1 计算当前节点到其邻节点的选择概率,依概率分布选择下一跳的节点,经过 k 步后得到一条长度为 k 的路径(line 2~line 5,算法 1).

Step 3: 检验生成路径里包含的不同节点的数量,若不同节点的数量少于 $\frac{2}{3}k$, 则放弃这一条路径(line 6~line 8,算法 1).

Step 4: 返回长度为 k 的节点序列用于语言模型生成节点的特征向量.

综上所述,假设复杂网络中节点数为 n,随机步长为 k,那么 k-step 启发式随机游走的时间复杂度为 $O(kn)$;若需要生成的随机游走序列数量为 m,那么时间复杂度为 $O(mkn)$.通常 $M<<m$,且 m 为常数,则 RandomWalkBy Weight 算法的时间复杂度为 $O(mkn)$.

当网络存在局部特殊拓扑结构,例如孤立连接或者孤立回路等极端情况时,k-step 随机游走过程会重复遍历相同的节点或形成局部回路,使路径中的节点重复率很高.因此,我们给随机游走路径中节点重复率设定一个阈值,若一条路径中不同节点的数量少于 $\frac{2}{3}k$, 则放弃生成的这条路径.k-step 随机游走过程中步长 k 的设定是至关重要的,过大的 k 值容易使随机游走过程陷入“重复陷阱”,提高了路径的弃用率并增加了算法的时间复杂度;过小的 k 值不能保证随机游走的覆盖效果.本文基于网络中社团平均直径较小的事实,通过在不同规模已知且有标记的网络上利用网格计算,得出 k 的经验最优值区间为[23,45].

3.2 分布式节点表达向量生成

3.2.1 统计语言模型

统计语言模型的目标是计算一个特定序列的词在语料库中出现的概率,假设 $W^n=(w_1,w_2,\ldots,w_n)$是由 n 个词 $w_1,w_2,\ldots,w_n$ 按顺序构成的一个句子,则句子的概率为词的联合概率 $\Pr(s)=\Pr(w_1,\ldots,w_{n-1},w_n)$,利用贝叶斯公式可以将其分解为

$$\Pr(s)=\Pr(w_1,...,w_{n-1},w_n)=\Pr(w_1)\Pr(w_1\mid w_2)...\Pr(w_n\mid w_1,...,w_{n-1})=\prod_{i=1}^{n}\Pr(w_i\mid Context_i),$$

其中,$Context_i$ 表示 w_i 的上下文, $\Pr(w_1)\Pr(w_1\mid w_2)...\Pr(w_n\mid w_1,...,w_{n-1})$ 就是统计语言模型的参数.

在统计语言模型中,假设语料库对应的字典的大小为 N,一个给定长度为 L 的句子,需要计算 L 个参数,考虑长度为 L 的任意句子有 N^L 种可能,而每种需要计算 L 个参数,通过计算 M^{N^L} 个参数,计算量巨大且存储这些信息内存开销也很大.为了快速计算模型参数,学术界提出多种模型,如 N-gram 模型、N-pos 模型和神经网络等方法.Bengio 等人[46]提出了一种基于词向量的神经概率语言模型.神经网络语言模型可以简单地归纳为:(1) 将单词映射到 m 维特征空间中;(2) 使用单词序列的对应向量集合作为输入表达单词序列的联合概率方程;(3) 同步学习单词的特征向量和概率函数.本文借助神经网络概率语言模型的思想,通过将整个复杂网络类比为语料库,复杂网络中的节点相当于语料库中的词,使用启发式随机游走生成节点序列 $W_v=(v_1,v_2,\ldots,v_i)$,节点序列把类比为统计语言模型中的词序列,对应的参数为 $\Pr(v_1)\Pr(v_1|v_2)\ldots\Pr(v_n|v_1,\ldots,v_{n-1})$,即目标函数为

$$\Pr(v_i\mid v_1,v_2,...,v_{n-1}) \tag{1}$$

我们希望得到一个能够表示节点间关系的特征向量,因此将要学习获得的各节点的向量表示为

$$\Phi: v\in V\rightarrow \mathbb{R}^{|V|\times d}.$$

Φ表示了复杂网络中节点之间潜在的相互关系,即节点的特征向量.目标函数由式(1)变为

$$\Pr(v_i\mid \Phi(v_1),\Phi(v_2),...,\Phi(v_{i-1})) \tag{2}$$

易知,随着随机游走长度 k 的增加,条件概率式(2)的分母有 $\sum_{i=1}^{k-1}i!$ 种情况,因此计算量将非常巨大.

3.2.2 SkipGram 模型

为了应对一般统计语言模型中条件概率计算量大的问题,文献[47,48]提出了一个放松的统计语言模型 SkipGram,该模型采用 w_i 预测其上下文 $w_{i-1},w_{i-2},w_{i+1},w_{i+2}$ 的概率,类似于 N-gram 模型规定了定长的“词语窗口”,w_i 的上下文仅由“词语窗口”内的词组成而不是整个句子;并且它不受“词语窗口”内的词序的约束,只需最大化地给定 w_i,不考虑“词语窗口”内词序的条件概率,不考虑任何其他先验知识.利用放松的统计语言模型方法,新的目标函数为

$$\text{minimize}_{\Phi}-\log\Pr(\Phi\{v_{i-w},...,v_{i+w}\}\mid\Phi\{v_i\}) \tag{3}$$

SkipGram 不考虑“词语窗口”内的词序的假设,对于复杂网络上的节点游走序列而言,节点的次序并不重要,所以,SkipGram 适合节点特征向量的学习.直观分析,网络游走序列相似的节点拥有相似的拓扑结构特征,也就

是网络游走序列相似的节点的特征向量应该相似,所以,可以通过优化目标函数式(3)得到节点的特征向量.

SkipGram 模型假设式(3)中的条件概率之间相互独立,得到:

$$\Pr(\Phi\{v_{i-w},...,v_{i-1},v_{i+1},v_{i+w}\}\,|\,\Phi\{v_i\})=\prod_{\substack{j=i-w\\ j\neq i}}^{i+w}\Pr(\Phi(v_j)\,|\,\Phi(v_i)) \tag{4}$$

SkipGram 模型的参数求解算法如算法 2 所示,算法 2 中节点 v 表示为 $\Phi(v)\in\mathbb{R}^d$,迭代遍历了节点序列中所有节点的窗口(line 1~line 2,算法 2).在给定节点 v_j 的情况下,最大化其节点序列中邻居节点的条件概率(line 3,算法 2),并据此更新节点向量的表示.算法 2 中有条件概率 $\Pr(\Phi(u_k)|\Phi(v_j))$,在神经概率语言模型中条件概率最朴素的形式为 $\Pr(\Phi(u_k)\,|\,\Phi(v_j))=\dfrac{e^{-E(\Phi(u_k),\Phi(v_j))}}{\sum_{i=1}^{V}e^{-E(\Phi(v_i),\Phi(v_j))}}$,其中,$E(\Phi(u_k),\Phi(v_j))=-(\Phi(u_k)\times\Phi(v_j))$ 为神经网络中常用的能量函数.易知,最里层的循环后验概率分母的计算量为 $O(|V|\times d)$,计算量非常大,该过程称为 softmax 归一化处理.为了提升算法效率,我们引入 Hierarchical Softmax 方法来计算后验概率.

3.2.3 Hierarchical Softmax 方法

由第 3.2.2 节可知,计算 $\Pr(\Phi(u_k)|\Phi(v_j))$的计算量非常大,主要是因为 softmax 归一化处理需要计算 $\sum_{i=1}^{V}e^{-E(\Phi(v_i),\Phi(v_j))}$,因此,使用 Hierarchical Softmax[49,50]方法来解决这一问题.另外,采用 Hierarchical Softmax 方法还能保证参数学习过程可以较快地收敛.

假设复杂网络中的每一个节点对应于 Huffman 树的一个叶子节点,将原来考虑整个复杂网络中所有节点的线性概率最大化问题转化为 Huffman 树由根节点到某一叶子节点的概率最大化问题,Huffman 树由启发式随机游走生成的节点序列中各节点出现的频率生成.假设复杂网络节点 u_k 为 Huffman 树中的一个叶子节点,从 Huffman 树的根节点到叶子节点 u_k 所有的非叶子节点为 $(b_0,b_1,...,b_{\lceil\log|V|\rceil-1})$,其中,$b_0$ 为根节点,$b_{\lceil\log|V|\rceil}$ 为 u_k,得到:

$$\Pr(\Phi(u_k)\,|\,\Phi(v_j))=\prod_{l=1}^{\lceil\log|V|\rceil}\Pr(\varphi(b_l)\,|\,\varphi(v_j)) \tag{5}$$

通过引入 Huffman 树表示 $\Pr(\Phi(u_k)|\Phi(v_j))$,将其转化成$\lceil\log|V|\rceil$个二分类,二分类函数使用 logistic 分类函数,得到:

$$\Pr(b_l\,|\,\Phi(v_j))=\frac{1}{1+e^{-\Phi(v_j)\times\Phi(b_l)}} \tag{6}$$

其中,b_l 为非叶子节点,$\Phi(b_l)\in\mathbb{R}^d$ 即非叶子节点的向量映射函数,$\varphi(b_l)$为类别向量.通过 Hierarchical Softmax 方法,我们将计算 $\Pr(\Phi(u_k)|\Phi(v_j))$的时间复杂度由 $O(|V|)$降低为 $O(\lceil\log|V|\rceil)$.

算法 2. SkipGram (Φ,W_{v_i},w).

输入:未更新的节点向量矩阵Φ, node sequence W_{v_i}, window size w;

输出:更新的节点向量矩阵Φ.

1: for each $v_j\in W_{v_i}$ do

2: for each $u_k\in W_{v_i}[j-w:j+w]$ do

3: $\Pr(u_k|\Phi(v_j))$

4: $\Phi=\Phi-\alpha\times\dfrac{\partial J}{\partial\Phi}$

5: end for

6: end for

3.2.4 训练模型参数

语言模型中的参数$\theta=\{\Phi,\varphi\}$,每一个参数的规模为 $\mathbb{R}^d$,参数可以通过随机梯度下降(SGD)[51]进行参数的学习.为了使用 SGD,需要求出参数的梯度.将公式(6)写成整体的形式,有:

$$\Pr(b_l\,|\,\Phi(v_j),\varphi_{l-1}^j)=[\sigma\Phi(v_j)^T\times\varphi_{l-1}^j(b_l)]^{1-b_l}\times[1-\sigma\Phi(v_j)^T\times\varphi_{l-1}^j(b_l)]^{b_l} \tag{7}$$

将式(7)代入式(5),再将式(5)代入式(4),对式(4)求对数似然可得:

$$L=\sum_{i\in V}\sum_{\substack{j=i-w\\ j\neq i}}^{i+w}\sum_{i=1}^{\lceil \log|V| \rceil}\{(1-b_l)\times[\sigma\Phi(v_j)^T\times\varphi_{l-1}^j(b_l)]+b_l\times[1-\sigma\Phi(v_j)^T\times\varphi_{l-1}^j(b_l)]\} \tag{8}$$

令:

$$L(v,j,l)=(1-b_l)\times[\sigma\Phi(v_j)^T\times\varphi_{l-1}^j(b_l)]+b_l\times[1-\sigma\Phi(v_j)^T\times\varphi_{l-1}^j(b_l)] \tag{9}$$

分别对式(9)求关于$\Phi(v_j)$和$\varphi_{l-1}^j(b_l)$的偏导数,得到式(10).

$$\begin{cases}\dfrac{\partial L(v,j,l)}{\partial\Phi(v_j)}=\left[1-b_l-\sigma(\Phi(v_j)^T\times\varphi_{l-1}^j(b_l))\right]\varphi_{l-1}^j(b_l)\\ \dfrac{\partial L(v,j,l)}{\partial\varphi_{l-1}^j(b_l)}=\left[1-b_l-\sigma(\Phi(v_j)^T\times\varphi_{l-1}^j(b_l))\right]\Phi(v_j)\end{cases} \tag{10}$$

由式(10)通过梯度下降法可得:

$$\begin{cases}\varphi_{l-1}^j(b_l)=\varphi_{l-1}^j(b_l)-\eta\times\left[1-b_l-\sigma(\Phi(v_j)^T\times\varphi_{l-1}^j(b_l))\right]\Phi(v_j)\\ \Phi(v_j)=\Phi(v_j)-\eta\times\sum\limits_{\substack{j=i-w\\ j\neq i}}^{i+w}\sum\limits_{i=1}^{\lceil \log|V| \rceil}\left[1-b_l-\sigma(\Phi(v_j)^T\times\varphi_{l-1}^j(b_l))\right]\varphi_{l-1}^j(b_l)\end{cases}$$

3.3 聚类算法

由于 *K*-Means 算法具有运行速度快、结构简单、可伸缩性等特点,所以我们使用 *K*-Means 算法对生成的节点特征向量进行聚类,得到复杂网络最终的划分结果.*K*-Means 聚类算法的时间复杂度是 $O(nXt)$,其中,n 表示复杂网络中的节点数,t 表示算法收敛之前的迭代次数,X 表示社团数目.*K*-Means 算法需要给定初始聚类中心,我们选择局部度中心节点[52]作为 *K*-Means 算法的聚类中心点.具体如算法 3 所示.

算法 3. *K*-Means (Φ,*k_nodes*).

输入:Matrix of vertex representations $\Phi\in\mathbb{R}^{|V|\times d}$; clustering centers *k_nodes*;

输出:Matrix of vertex's group representations $group\in\mathbb{R}^{|V|\times 1}$.

```
while k_nodes≠new_k_nodes do
    for other_node in Φ, k_nodes do
        for center_node in k_node do
            calculate the cosine distance between other_node and center_node
        end for
        other_node belongs to min cosine distance's center
    end for
    calculate new_k_nodes through the average of current group nodes
end while
```

3.4 CDNEV算法整体流程

利用启发式随机游走算法,对每个节点生成固定长度的上下文节点序列,然后利用 SkipGram 模型生成分布式节点表示向量,最后利用特征向量进行聚类,得到复杂网络中社团划分的结果,这是 CDNEV 算法的核心过程,CDNEV 算法的具体步骤描述如算法 4 所示.

算法核心步骤可以分为 7 步.第 1 步,也就是算法第 1 行,初始化各节点向量,得到矩阵 $\Phi\in\mathbb{R}^{|V|\times d}$,通常情况下,每个节点向量的初始化是使用随机数的方法来实现;第 2 步,算法第 2 行,依据算法 1 中完全随机游走产生的随机节点序列中节点的被遍历频率创建 Huffman 树;第 3 步,进入启发式游走迭代,算法第 3 行开始,算法第 4 行,生成无序的复杂网络节点序列;第 4 步,从算法的第 5 行开始,对序列中的每个节点进行启发式随机游走,生成长

度为 k 的节点序列,算法第 6 行执行;算法第 7 行,采用 SkipGram 算法对每个节点向量进行优化并更新;第 5 步,算法的第 10 行,计算得到每个节点的分布式实向量,其维度采用 d 表示;第 6 步,算法的第 11 行,选择局部中心度大于网络中所有节点的局部平均中心度,且处于网络中前 10%的 X 个局部度中心节点作为聚类的中心节点;第 7 步,算法 4 的第 12 行,采用 K-Means 算法进行聚类,得到社团划分结果.

算法 4. CDNEV (G,w,k,d,r).

输入:Graph $G(V,E)$; window size w; step size k; vector size d; walk per vertex r;

输出:Matrix of vertex's group representations $group \in \mathbb{R}^{|V|\times 1}$.

1: Initialization: Sample Φ from $U|V|^{\times d}$

2: Build a Huffman Tree T from V

3: for i=0 to r do

4: O=*Suffle*(V)

5: for each $v_i \in O$ do

6: $W_{v_i} = RandomWalkByWeight(G, v_i, k)$

7: SkipGram (Φ,W_{v_i},w)

8: end for

9: end for

10: matrix of vertex representations $\Phi \in \mathbb{R}^{|V|\times d}$

11: select *k_nodes* with high local degree central

12: K-Means (Φ,*k_nodes*)

4 实验结果与分析

我们从优化方法和启发式方法中选取 10 种经典的社团发现算法进行对比分析和比较,包括 GN 算法[25]、Newman 快速算法(FG)[16]、Leading eigenvector 算法(LE)[53]、Infomap 算法(IM)[54]、Label propagation 算法(LPA)[27]、Spinglass 算法(SG)[55]、Walktrap 算法(WT)[56]、Louvain 算法(LV)[26]、GCE 算法[22]和 LFM 算法[21]和 SHRINK 算法(SK)[24].

为了更客观地衡量本文算法的性能,我们也和 DeepWalk(DW)算法[3]、node2vec(N2V)算法进行了比较.在用 DeepWalk 算法继续进行社团划分时,我们采取了和本文算法一致的策略,也就是说,在学习到节点向量的基础上用 K-Means 算法进行聚类,从而得到社团划分的结果.

我们采用社团划分领域常用的测试数据集进行实验,数据集包括真实网络数据集和模拟网络数据集.真实网络数据集包括 Zachary 空手道俱乐部成员关系网络、宽吻海豚网络、美国政治书籍网络、美国大学生橄榄球网络、西班牙大学 email 通信网络和网络科学领域科学家合作网络.模拟数据集主要由 LFR 基准图组成,选择不同的参数生成不同的 LFR 基准图.

在进行实验时,我们参考了 word2vec 对于节点向量长度的人为设定,且针对大规模网络,将节点嵌入实向量的维度设为 40.

4.1 算法结果评价指标

向量的维度设为 40.在实验中我们发现,准确率(precision)、召回率(recall)、F 指标(F1)和模块度(modularity)最能刻画算法划分结果与真实结果的差异,为了科学地衡量不同算法的性能,我们采用上述 4 种评价指标对不同算法的效果进行评价.

(1) 准确率(precision,简称 P)

准确率代表社团划分结果中正确节点数量占总节点数量的比例,计算方法为

$$P=\frac{|L_{\text{result}}|\cap|L_{\text{true}}|}{|L_{\text{result}}|},$$

其中,L_{result} 表示算法得到的社团,L_{ture} 表示真实社团.

(2) 召回率(recall,简称 R)

召回率反映了真实社团中被正确划分出的节点的比例,计算方法为

$$R=\frac{|L_{\text{result}}|\cap|L_{\text{true}}|}{|L_{\text{true}}|}.$$

(3) F 指标($F1$)

$F1$ 评价指标是对 P 值和 R 值的综合,计算公式为

$$F1=\frac{2\times P\times R}{P+R},$$

其中,P 为准确率,R 为召回率.

(4) 模块度(modularity)

模块度用来评价无标记网络中的社团划分结果,又称为 Q 函数.Q 函数为社团内实际连接数目与随机连接情况下社团内期望连接数目之差,用来对网络中社团的整体质量做出一个定量的评价.计算方法如下:

$$Q=\frac{1}{2m}\sum_{ij}\left(A_{ij}-\frac{k_ik_j}{2m}\right)\sigma(c_i,c_j),$$

其中,k_i 和 k_j 表示节点的度;c_i 表示节点 i 所属社团;m 表示网络的总边数.当 c_i=c_j 时,$\sigma(c_i,c_j)$=1,否则为 0.

(5) 标准化互信息(normalized mutual information,简称 NMI)

标准化互信息用来评价聚类效果.其中,U 对应标准结果,V 对应聚类的预测结果,计算方法为

$$NMI(U,V)=\frac{MI(U,V)}{\sqrt{H(U)H(V)}},$$

其中,

$$MI(U,V)=\sum_{i=1}^{|U|}\sum_{j=1}^{|V|}P(i,j)\log\left(\frac{P(i,j)}{P(i)P'(j)}\right),\ H(U)=\sum_{i=1}^{|U|}P(i)\log(P(i)),\ H(V)=\sum_{j=1}^{|V|}P'(j)\log(P(j)).$$

(6) 调整兰德系数(adjusted rand index,简称 ARI)

调整兰德系数用来评价聚类结果的准确程度,计算方法为

$$ARI=\frac{RI-E[RI]}{\max(RI)-E[RI]},$$

其中,

$$RI=\frac{a+b}{C_2^{n_{samples}}},$$

其中,a 表示标准结果与预测结果相同的样本数量,b 表示标准结果与预测结果不相同的样本数量.

4.2 真实网络数据集实验

实验采用的真实网络统计信息见表 2.将 CDNEV 和其他基准的社团划分算法应用于真实网络数据集,用每种算法对不同数据集进行划分.对于有标记网络,因为其标注了每个节点所属社团,所以我们计算各算法划分结果的 P、R 和 $F1$ 指标,并将它们作为定量比较各算法性能的重要依据;对于未标注网络,即不确定社团结构的网络,因其并没有对每个节点归属社团进行标注,无法采用 P、R 和 $F1$ 指标进行比较,本文采用被学术界广泛采用的模块度对算法划分结果进行比较.对于规模较小的标记网络,可以人工识别标准结果与社团划分结果的对应关系,对于规模较大的标记网络,则无法人工识别此对应关系,所以,使用 ARI 指标来对社团划分结果进行度量.各算法在真实网络中的划分结果见表 3.

由表 3 可知,在 Karate 网络中,CDNEV 算法是唯一能够将各节点划分到其所属社团的算法,在其他有标注

数据集中,CDNEV 算法的表现无论在准确率、召回率单个指标上,还是在综合评价指标 $F1$ 上,均优于对比算法.我们在 4 个数据集上,比较了算法的平均指标.

Table 2 Real networks used in the experiment

表 2 实验中用到的真实网络

网络	缩写	$\|V\|$	$\|E\|$	标记网络
Zachary's karate club[42]	Karate	34	78	√
Dolphin social network[43]	Dolphin	62	159	√
American college football[8]	Football	115	616	√
Books about US politics[10]	Polbooks	105	441	√
Email communicatin network[44]	Email	1 133	5 451	×
com-DBLP[57]	DBLP	317 080	1 049 866	√

Table 3 Comparations of CDNEV and baseseline algorithms' community detection on real networks

表 3 CDNEV 算法与其他算法在真实网络中划分结果的比较

Data		GN	FG	IM	SG	WT	LV	LE	LPA	GCE	LFM	SK	DW	N2V	CDNEV
Karate	*P*	0.971	0.971	0.971	**1.000**	0.941	0.971	**1.000**	0.971	0.971	0.882	0.941	0.971	0.971	**1.000**
	R	0.516	0.722	0.761	0.563	0.411	0.545	0.533	0.761	0.972	0.908	0.956	0.972	0.972	**1.000**
	*F*1	0.674	0.828	0.853	0.720	0.572	0.698	0.695	0.853	0.972	0.895	0.948	0.972	0.972	**1.000**
Dolphins	*P*	**0.984**	0.935	0.968	0.968	0.952	0.935	0.952	0.968	0.871	0.935	0.935	0.645	0.968	**0.984**
	R	0.619	0.619	0.478	0.489	0.603	0.452	0.406	0.549	0.438	0.817	0.803	0.548	0.819	**0.985**
	*F*1	0.745	0.745	0.639	0.650	0.738	0.610	0.569	0.700	0.583	0.872	0.864	0.593	0.887	**0.984**
Football	*P*	0.835	0.574	**0.930**	0.878	0.870	0.870	0.626	0.800	0.852	0.713	0.922	0.270	0.852	0.922
	R	0.954	**0.992**	0.929	0.968	0.972	0.977	0.887	0.977	0.917	0.976	0.883	0.267	0.887	0.925
	*F*1	0.890	0.727	**0.930**	0.921	0.918	0.920	0.734	0.879	0.883	0.824	0.902	0.268	0.869	0.923
Polbooks	*P*	**0.857**	0.838	0.848	0.848	0.848	0.848	0.848	0.848	0.762	0.838	0.887	0.819	0.838	0.848
	R	0.763	0.769	0.596	0.557	0.761	0.627	0.599	0.763	0.746	0.793	0.879	0.776	0.746	**0.944**
	*F*1	0.807	0.802	0.700	0.672	0.802	0.721	0.702	0.803	0.754	0.815	0.883	0.797	0.789	**0.893**

$F1$ 指标值最低的是 DW 算法,为 0.657 5,最高的是 LFM 算法,为 0.851 5,CDNEV 算法平均 $F1$ 则为 0.95,明显高于其他算法.相对于其他算法,CDNEV 算法的 $F1$ 指标平均提高了 19%.

在一些特殊情况下,如 SG 算法和 LE 算法,在 Karate 数据集上的准确率也可以达到 100%,但这些算法是将一个大社团中的节点划分到多个子社团,这种"过度划分"导致它们的召回率很低,所以最终的 $F1$ 值较低.由第 3.5 节易知,由于 SG 算法是基于贪心迭代算法的一种改进,因此,SG 算法相较于 CDNEV 算法,其时间复杂度较高. CDNEV 算法与其他算法在 E-mail 网络上的模块度比较可见表 4.

Table 4 Comparations of CDNEV and baseline algorithms' modularity on E-mail networks

表 4 CDNEV 算法与其他算法在 E-mail 网络上的模块度比较

Data	GN	FG	IM	SG	WT	LV	LE	LPA	GCE	LFM	SK	DW	N2V	CDNEV
M	0.532	0.506	0.52	**0.579**	0.53	0.54	0.488	0.532	0.436	0.127	0.527	0.513	0.548	**0.553**

DBLP 网络是标记网络,且网络规模较大,各算法在社团划分上的 ARI 指标见表 5,CDNEV 算法的 ARI 指标与 IM、N2V 等算法基本持平,高于其他算法.验证了 CDNEV 算法在较大规模网络中的可扩展性.

Table 5 Comparations of CDNEV and baseline algorithms' ARI on DBLP networks

表 5 CDNEV 算法与其他算法在 DBLP 网络上的 ARI 指标比较

Data	GN	FG	IM	SG	WT	LV	LE	LPA	GCE	LFM	SK	DW	N2V	CDNEV
ARI	0.314	0.353	0.396	**0.412**	0.372	0.393	0.362	0.401	0.339	0.218	0.407	0.357	**0.413**	**0.413**

综合无标记和有标记网络上的实验结果可知,相较于其他算法,CDNEV 算法在真实网络中的划分效果优于其他算法.

4.3 模拟网络数据集实验

为了进一步评估这些算法的性能,我们采用 LFR[58]人工合成网络标准网络来进行实验,LFR 网络中节点的度分布及社团的规模分布均为幂律分布,使其更接近真实网络.本次实验主要通过设置以下参数来生成所需的

模拟复杂网络.

模拟网络节点总数为 n;模拟网络的节点平均度为 $k(P)$;模拟网络节点最大度为 $k_{max}(P_{max})$.另外,还需要通过实验分析、比较拓扑混合参数μ的作用,μ表示模拟网络中社团内节点与社团外部节点连接的边数占节点总边数的比例,μ越大,说明网络结构越不明显;具有指数分布形式的度分布的参数ε_1,模拟网络节点度分布服从幂指数为ε_1的幂律分布;LFR 网络的社区规模服从指数分布,其参数为ε_2;最小社区规模为 c_{min},指定最小社区的节点数;最大社区规模为 c_{max},指定最大社区的节点数.

按照文献[59]中的实验设计建议,对 LFR 基准网络的参数设置如下.

(1) 网络规模 n 取值为 1 000;

(2) 最小社团规模 c_{min} 取值为 10 或 20;

(3) 混合参数μ从 0.05 变化到 0.7,间隔为 0.05.

(4) 我们保持其他参数不变,即节点的平均度 $k(P)$为 20;最大度 $k_{max}P_{max}$ 为 2.5 倍 $k(P)$;最大社团规模 c_{max} 为 5 倍 c_{min};节点度与社团规模的幂律分布指数分别为ε_1=–2,ε_2=–1.

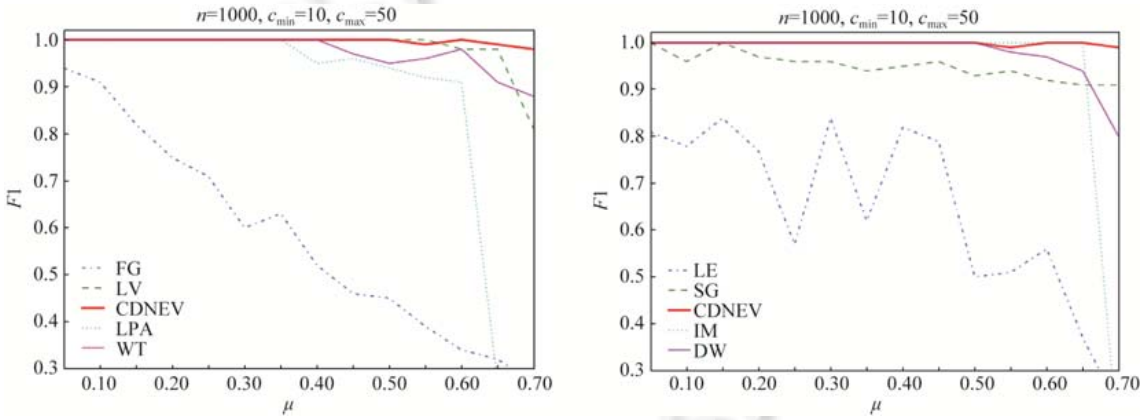
我们通过设置 LFR,模拟不同参数,进行了 3 组实验.

实验 1:设 n=1000,c_{min}=10,c_{max}=50,混合参数μ为变化量,各算法对应的 $F1$ 值如图 1 所示.

实验 2:设 n=1000,c_{min}=20,c_{max}=100,混合参数μ为变化量,各算法对应的 $F1$ 值如图 2 所示.

实验 3:设μ=0.65,c_{min}=10,c_{max}=50,社团规模 n 为变化量,各算法对应的 $F1$ 值如图 3 所示.

对实验 1 和实验 2,我们生成了 2 个网络规模相同、但网络结构不同的 LFR 模拟网络,通过分析不同算法随着混合参数μ取不同值的划分效果.为了更进一步说明 CDNEV 算法和其他算法的性能差异,实验 3 中我们固定μ=0.65,把网络规模从 1 000 按间隔为 1 000 逐步提升到 10 000,然后对比分析不同算法性能随着网络规模变化的情况.



(a) (b)

Fig.1 The variation of different algorithms' $F1$ with different μ on base networks, n=1000, c_{min}=10, c_{max}=50

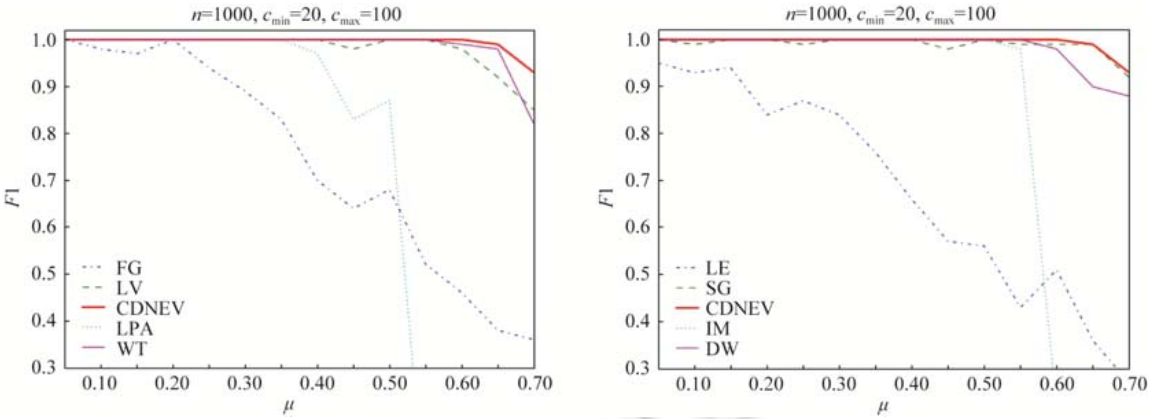
图 1 不同算法在基准网络上 $F1$ 值随μ的变化结果,n=1000,c_{min}=10,c_{max}=50

图 1 表示在 n=1000,c_{min}=10,c_{max}=50;图 2 表示在 n=1000,c_{min}=20,c_{max}=100,分别在混合参数μ取不同值时,得到对应的 $F1$ 值.图 1 和图 2 的 2 个子图是 CDNEV 和不同算法的比较曲线.图 1 和图 2 的横轴表示μ值,纵轴表示 $F1$ 值.

从图 1 和图 2 可以看出,在混合参数μ取值较小时(μ<0.2),10 种算法所表现出来的 $F1$ 值差别不明显, 特别是在 c_{min}=20,c_{max}=100 的情况下.但是,随着混合参数μ值的增大,网络中社区结构开始变得模糊化,不同算法之间的性能差异开始变大.由图 1 和图 2 可知,CDNEV 对应的 $F1$ 值一直高于除 WT 以外的其他算法,并且与 WT 的

差距最多不超过 0.02.基于随机游走的 WT 算法的时间复杂度主要依赖于网络中边的数目,时间复杂度为 $O(mn^2)$,而 CDNEV 的时间复杂度仅为 $O(n\log n)$,因此本文提出的算法效率远高于 WT 算法.

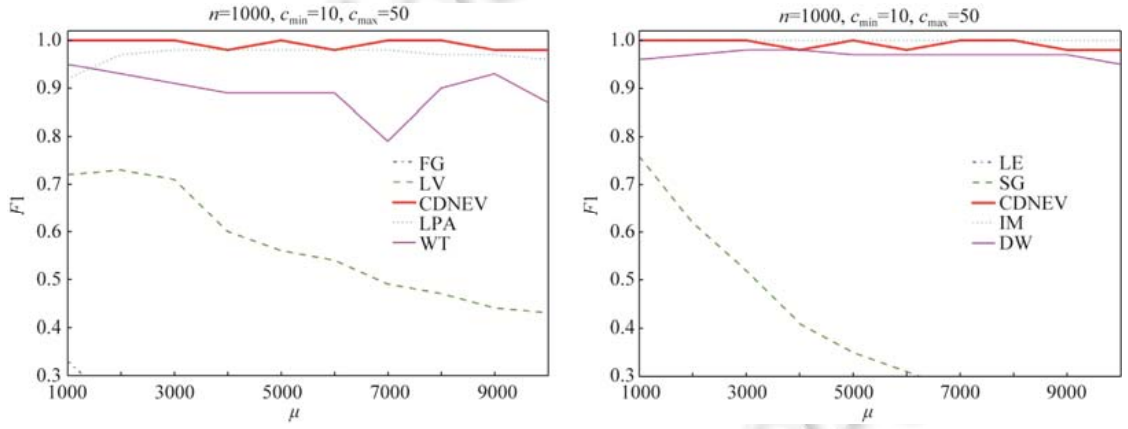
图 3 给出了 CDNEV 算法和其他算法在不同网络规模下的 $F1$ 指标变化曲线.



(a) (b)

Fig.2 The variation of different algorithms' $F1$ with different μ on base networks, n=1000, c_{min}=20, c_{max}=100

图 2 不同算法在基准网络上 $F1$ 值随μ的变化结果,n=1000,c_{min}=20,c_{max}=100



(a) (b)

Fig.3 The variation of different algorithms' $F1$ with different network scales on base networks

图 3 不同算法在基准网络规模变化下的 $F1$ 值

如图 3 所示,可以发现,随着网络规模的扩大,不同算法的性能差别同样逐渐加大.原本在网络规模为 1 000 时表现最好的 WT 算法,随着网络规模的扩大,性能逐渐变差.而本文所提出的 CDNEV 算法依然保持较高的水准,IM 算法和 CDNEV 算法在网络规模扩大时性能差异不明显,IM 算法和 CDNEV 算法优于其他算法,但是 IM 算法的时间复杂度为 $O(n(m+n))$,也高于 CDNEV 算法的 $O(n\log n)$,结果说明,CDNEV 算法能够应用于大规模网络上的社团划分.

随着模拟网络节点和边的数量的增加,网络结构更加复杂,导致算法在真实网络和模拟网络上的结果有所下降.但从实验结果中可以清晰地看出,随着网络规模的扩大,所有算法的性能都在下降,CDNEV 算法表现得相对于其他算法依然有一定优势.

4.4 节点向量的有效性分析

为了深入分析 CDNEV 算法生成向量的有效性,我们对 Karate 网络中的节点进行启发式随机游走,然后采用节点向量表示算法对每个节点生成一个二维的分布式向量.

为了验证节点二维分布式向量的效果,我们用二维向量做散点图,如图 4 所示.图 4 中的每个点代表一个节点,不同的颜色表示节点所属社团.

从图 4 可以看出,虽然只用了二维向量,放弃了部分维度上的信息,但用二维向量作为节点距离,然后对节点聚类,依然能够区分社团结构.从散点图可以看出,只有 1 个蓝色的节点靠近红色的社团、1 个红色的节点靠近蓝色的社团,社团之间的重叠区域较少.随着维度的增加,社团内部的节点距离将会更加紧密,这样保证了 CDNEV 算法生成的节点分布式向量能够有效地表示节点在社团结构上的特性.

综合真实网络和模拟网络上的实验结果分析,CDNEV 算法具有简单、有效的特点,适合于在大型复杂的网络上进行社团划分.

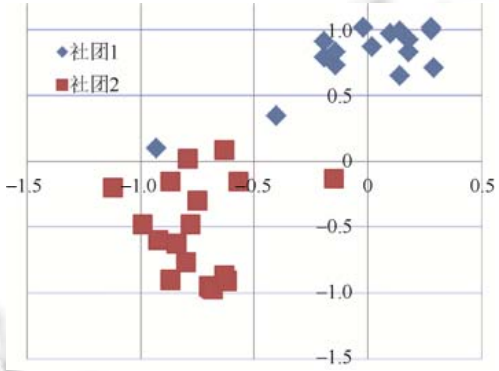


Fig.4 Relationship of 2D vector scatter diagram and community structure

图 4 二维向量散点图与社团结构的关系

5 总结与未来工作

本文提出一种结合自然语言处理方法与聚类方法的社团划分算法 CDNEV.CDNEV 算法首先通过启发式的随机游走算法生成节点上下文序列,然后将节点序列类比为自然语言处理中的词序列,使用 SkipGram 算法学习能够代表每个节点的节点特征向量,在学习过程中使用 Hierarchical Softmax 方法加快特征向量的学习速率,最后依据网络中的核心节点,使用 *K*-Means 聚类方法得到最后复杂网络的社团结构.CDNEV 算法与其他经典的社区发现算法在多种真实网络和模拟基准网络数据集上进行了比较,实验结果表明,算法在大多数数据集上具有一定优势,不论在有标记网络中的 *F*1 值,还是在无标记网络中的模块度值都处于较高水准,同时,算法复杂度较小,执行效率快,说明 CDNEV 算法能够适用于大规模复杂网络的社团划分任务,而且能够保持较高的精度.

在大型网络上存在重叠社团结构,如何构造满足重叠等复杂社区结构的节点向量是一个值得研究的方向.CDNEV 模型目前仅使用了网络的结构信息,在未来的工作中,还将引入节点上的文本标记等其他信息.另外,研究算法的并行化处理,使得算法能够应用于大规模网络也是值得研究的方向.对于不同的社团划分算法,还应考虑统一开发的实验语言与运行平台,以便更好地测试不同算法在实际应用中的执行效率.

References:

[1] Newman MEJ, Watts DJ. Renormalization group analysis of the small-world network model. Physics Letters A, 1999,263(4): 341–346.

[2] Shao F, Jiang GP. Optimal traffic routing strategy based on community structure. Acta Physica Sinica, 2011,60(7):078902 (in Chinese with English abstract).

[3] Perozzi B, Al-Rfou R, Skiena S. Deepwalk: Online learning of social representations. In: Proc. of the 20th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM, 2014. 701–710.

[4] Bedi P, Sharma C. Community detection in social networks. Wiley Interdisciplinary Reviews Data Mining & Knowledge Discovery, 2016,496-500(3).

[5] Fortunato S, Hric D. Community detection in networks: A user guide. Physics Reports, 2016,659:1–44.

[6] Newman MEJ. Modularity and community structure in networks. Proc. of the National Academy of Sciences, 2006,103(23): 8577–8582.

[7] Shiga M, Takigawa I, Mamitsuka H. A spectral clustering approach to optimally combining numericalvectors with a modular network. In: Proc. of the 13th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM, 2007. 647–656.

[8] Fu LD, Gao L, Ma XK. A centrality measure based on spectral optimization of modularity density. Science China Information Sciences, 2012,42(5):550–560 (in Chinese with English abstract).

[9] Riolo MA, Newman MEJ. First-principles multiway spectral partitioning of graphs. Journal of Complex Networks, 2014,2(2): 121–140.

[10] Liu R, Feng S, Shi R, *et al*. Weighted graph clustering for community detection of large social networks. Procedia Computer Science, 2014,31:85–94.

[11] Wang Z, Chen Z, Zhao Y, *et al*. A community detection algorithm based on topology potential and spectral clustering. The Scientific World Journal, 2014,2014:329325. [doi: 10.1155/2014/329325]

[12] Jin H, Wang S, Li C. Community detection in complex networks by density-based clustering. Physica A: Statistical Mechanics and its Applications, 2013,392(19):4606–4618.

[13] Lin W, Kong X, Yu PS, *et al*. Community detection in incomplete information networks. In: Proc. of the 21st Int'l Conf. on World Wide Web. ACM, 2012. 341–350.

[14] Gennip Y, Hunter B, Ahn R, *et al*. Community detection using spectral clustering on sparse geosocial data. SIAM Journal on Applied Mathematics, 2013,73(1):67–83.

[15] Kernighan BW, Lin S. An efficient heuristic procedure for partitioning graphs. Bell System Technical Journal, 1970,49(2): 291–307.

[16] Newman MEJ. Fast algorithm for detecting community structure in networks. Physical Review E, 2004,69(6):066133.

[17] Pan Y, Li DH, Liu JG, *et al*. Detecting community structure in complex networks via node similarity. Physica A: Statistical Mechanics and its Applications, 2010,389(14):2849–2857.

[18] Zanjani AAH, Darooneh AH. Finding communities in linear time by developing the seeds. Physical Review E, 2011,84(3):036109.

[19] Wang XY, Zhao ZX. Partitioning community structure in complex networks based on node dependent degree. Acta Physica Sinica, 2014,63(17):178901 (in Chinese with English abstract).

[20] Clauset A. Finding local community structure in networks. Physical review E, 2005,72(2):026132.

[21] Lancichinetti A, Fortunato S, Kertész J. Detecting the overlapping and hierarchical community structure in complex networks. New Journal of Physics, 2009,11(3):033015.

[22] Lee C, Reid F, McDaid A, *et al*. Detecting highly overlapping community structure by greedy clique expansion. arXiv Preprint arXiv:1002.1827, 2010.

[23] Xu X, Yuruk N, Feng Z, *et al*. SCAN: A structural clustering algorithm for networks. In: Proc. of the ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM, 2007. 824–833.

[24] Huang J, Sun H, Han J, *et al*. SHRINK: A structural clustering algorithm for detecting hierarchical communities in networks. In: Proc. of the ACM Conf. on Information and Knowledge Management, CIKM 2010. Toronto: DBLP, 2010. 219–228.

[25] Girvan M, Newman MEJ. Community structure in social and biological networks. Proc. of the National Academy of Sciences, 2002, 99(12):7821–7826.

[26] Blondel VD, Guillaume JL, Lambiotte R, *et al*. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment, 2008,(10):P10008.

[27] Raghavan UN, Albert R, Kumara S. Near linear time algorithm to detect community structures in large-scale networks. Physical Review E, 2007,76(3):036106.

[28] Sun H, Liu J, Huang J, *et al*. CenLP: A centrality-based label propagation algorithm for community detection in networks. Physica A: Statistical Mechanics & Its Applications, 2015,436:767–780.

[29] Tsourakakis C, Gkantsidis C, Radunovic B, *et al*. Fennel: Streaming graph partitioning for massive scale graphs. In: Proc. of the 7th ACM Int'l Conf. on Web Search and Data Mining. ACM, 2014. 333–342.

[30] Yuan C, Chai Y. Method for local community mining in the complex networks. Acta Automatica Sinica, 2014,40(5):921–934 (in Chinese with English abstract).

[31] Albert R, DasGupta B, Mobasheri N. Topological implications of negative curvature for biological and social networks. Physical Review E, 2014,89(3):032811.

[32] Gan WY, Nan HE, Li DY, *et al*. Community discovery method in networks based on topological potential. Ruan Jian Xue Bao/Journal of Software, 2009,20(8):2241–2254 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/3318.htm [doi: 10.3724/SP.J.1001.2009.03318]

[33] Wu F, Huberman BA. Finding communities in linear time: A physics approach. The European Physical Journal B-Condensed Matter and Complex Systems, 2004,38(2):331–338.

[34] He D X, Xu Z, Zuo W, *et al*. Community mining in complex networks—Clustering combination based genetic algorithm. Acta Automatica Sinica, 2010,36(8):1160–1170 (in Chinese with English abstract).

[35] Okamoto H. Local detection of communities by attractor neural-network dynamics. In: Artificial Neural Networks. Springer Int'l Publishing, 2015. 115–125.

[36] Zhang ZY. Community structure detection in social networks based on dictionary learning. Science China (Information Sciences), 2011,41(11):1343–1355 (in Chinese with English abstract).

[37] Tang L, Liu H. Scalable learning of collective behavior based on sparse social dimensions. In: Proc. of the 18th ACM Conf. on Information and Knowledge Management. ACM, 2009. 1107–1116.

[38] Tang J, Qu M, Wang M, *et al*. Line: Large-scale information network embedding. In: Proc. of the 24th Int'l Conf. on World Wide Web. Int'l World Wide Web Conferences Steering Committee, 2015. 1067–1077.

[39] Cao S, Lu W, Xu Q. GraRep: Learning graph representations with global structural information. In: Proc. of the 24th ACM Int'l on Conf. on Information and Knowledge Management. ACM, 2015. 891–900.

[40] Wang DX, Cui P, Zhu WW. Structural Deep Network Embedding. In: Proc. of the 22nd ACM SIGKDD Int'l Conf. 2016. 1225–1234. [doi: 10.1145/2939672.2939753]

[41] Li J, Ritter A, Jurafsky D. Learning multi-faceted representations of individuals from heterogeneous evidence using neural networks. arXiv Preprint arXiv:1510.05198, 2015.

[42] Zhou N, Zhao WX, Zhang X, *et al*. A general multi-context embedding model for mining human trajectory data. IEEE Trans. on Knowledge & Data Engineering, 2016,28(8):1945–1958.

[43] Yang C, Liu Z, Zhao D, *et al*. Network representation learning with rich text information. In: Proc. of the 24th Int'l Joint Conf. on Artificial Intelligence. Buenos Aires, 2015. 2111–2117.

[44] Chen J, Zhang Q, Huang X. Incorporate group information to enhance network embedding. In: Proc. of the ACM Int'l on Conf. on Information and Knowledge Management. ACM, 2016. 1901–1904.

[45] Tang J, Qu M, Mei Q. PTE: Predictive text embedding through large-scale heterogeneous text networks. In: Proc. of the 21st ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM, 2015. 1165–1174.

[46] Bengio Y, Ducharme R, Vincent P, *et al*. A neural probabilistic language model. The Journal of Machine Learning Research, 2003, 3:1137–1155.

[47] Mikolov T, Chen K, Corrado G, *et al*. Efficient estimation of word representations in vector space. arXiv Preprint arXiv:1301.3781, 2013.

[48] Mikolov T, Sutskever I, Chen K, *et al*. Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems. 2013. 3111–3119.

[49] Mnih A, Hinton GE. A scalable hierarchical distributed language model. In: Advances in Neural Information Processing Systems. 2009. 1081–1088.

[50] Morin F, Bengio Y. Hierarchical probabilistic neural network language model. In: Proc. of the Int'l Workshop on Artificial Intelligence and Statistics. 2005. 246–252.

[51] Bottou L. Stochastic gradient learning in neural networks. Proc. of Neuro-Nımes, 1991,91(8).

[52] Chen Q, Wu TT, Fang M. Detecting local community structures in complex networks based on local degree central nodes. Physica A: Statistical Mechanics and Its Applications, 2013,392(3):529–537.

[53] Newman MEJ. Finding community structure in networks using the eigenvectors of matrices. Physical Review E, 2006,74(3): 036104.

[54] Rosvall M, Axelsson D, Bergstrom CT. The map equation. The European Physical Journal Special Topics, 2010,178(1):13–23.

[55] Reichardt J, Bornholdt S. Statistical mechanics of community detection. Physical Review E, 2006,74(1):016110.

[56] Pons P, Latapy M. Computing communities in large networks using random walks. In: Computer and Information Sciences-ISCIS 2005. Berlin, Heidelberg: Springer-Verlag, 2005. 284–293.

[57] Yang J, Leskovec J. Defining and evaluating network communities based on ground-truth. Knowledge & Information Systems, 2015,42(1):181–213.

[58] Lancichinetti A, Fortunato S, Radicchi F. Benchmark graphs for testing community detection algorithms. Physical Review E, 2008, 78(4):046110.

[59] Lancichinetti A, Fortunato S. Community detection algorithms: A comparative analysis. Physical Review E, 2009,80(5):056117.

附中文参考文献:

[2] 邵斐,蒋国平.基于社团结构的负载传输优化策略研究.物理学报,2011,60(7):078902.

[8] 付立东,高琳,马小科.基于社团检测的复杂网络中心性方法.中国科学(信息科学),2012,42(5):550–560.

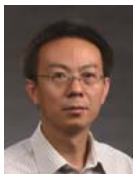
[19] 王兴元,赵仲祥.基于节点间依赖度的社团结构划分方法.物理学报,2014,63(17):178901.

[30] 袁超,柴毅.复杂网络的局部社团结构挖掘算法.自动化学报,2014,40(5):921–934.

[32] 淦文燕,赫南,李德毅,王建民.一种基于拓扑势的网络社区发现方法.软件学报,2009,20(8):2241–2254. http://www.jos.org.cn/1000-9825/3318.htm [doi: 10.3724/SP.J.1001.2009.03318]

[34] 何东晓,周栩,王佐,等.复杂网络社区挖掘——基于聚类融合的遗传算法.自动化学报,2010,36(8):1160–1170.

[36] 张忠元.基于字典学习的网络社团结构探测算法.中国科学(信息科学),2011,41(11):1343–1355.

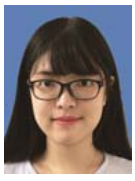


韩忠明(1972－),男,山西吕梁人,博士,教授,CCF 专业会员,主要研究领域为社会网络,数据挖掘,大数据处理.

郑晨烨(1994－),女,学士,主要研究领域为社交网络挖掘.

刘雯(1992－),男,学士,主要研究领域为社交网络挖掘.

谭旭升(1990－),男,学士,主要研究领域为社交网络挖掘.



李梦琪(1993－),女,学士,主要研究领域为深度学习,自然语言处理.

段大高(1976－),男,博士,副教授,CCF 专业会员,主要研究领域为异构数据挖掘,大数据处理,社会网络.