基于云计算的文献合作网络的社团划分及演化分析*

□ 杜雅红 白云龙 吴斌 / 北京邮电大学智能通信软件与多媒体北京市重点实验室 北京 100876

摘要：随着各领域科学研究的开展，文献数据与日俱增，关于文献数据的更深入的研究对于科研对象的评价与趋势分析具有重要意义。同时，随着分析方法的成熟和应用场景的延伸，文献的分析带给研究人员的一个挑战是如何在超大规模数据（PB级）中进行有效的挖掘。工业界和学术界越来越倾向于使用基于分布式密集数据计算模型，如MapReduce等，来进行大规模数据挖掘。文章在云计算平台上实现了文献数据的社团发现算法，并对学术会议的演化进行了分析。通过分析，可以清晰地看到活跃在学术会议中的科研合作团队的核心科研工作者，这对进一步了解学术会议研究方向及动态具有很强的指导性作用。

关键词：文献数据，图挖掘，社团划分，演化分析，云计算
DOI：10.3772/j.issn.1673—2286.2010.08.007

1 导言

随着科学研究、通讯技术、IT技术的快速发展，各种信息充斥着人们的生活。随之而来的是越来越多的数据以及隐藏在数据中的各种各样的关系，比如企业用户行为数据、通话数据、文献数据、网上虚拟社区的关系数据、医学试验的各种繁琐数据等等。其中，对于引领社会科学发展的科学文献网络，人们希望从中探知：当今最热门的学科方向是什么？哪些传统学科的研究已经走向衰败？对于某一领域哪个研究者或者哪个研究团队是研究的核心力量？各个团体间的合作模式是怎样的？人们已经对文献数据进行了很多方面的分析，比如社团发现、社会影响性分析、合作网络演化分析等。这些分析帮助我们获知各个领域的发展动态、资源分布、交叉关联关系以及科研团队之间的合作关系等信息，以便很好地理清科技发展的脉络，合理地分配各种资源，同时有助于促进科技走向市场化。

然而随着信息产业和整个社会的发展，越来越多的数据被收集起来，仅成立10年的国家科技图书文献中心（NSTL）的数据总量已上亿条，国内知名的数字期刊出版企业的数据规模都达到T级（10^{12}）。因此，如何从海量数据中发现特定知识，如何高效地处理海量数据几乎是任何一个信息分析机构要面对的问题。一般来说，并行计算能够加快处理速度，而并行计算也有多种选择，例如，在Web Service基础上的网格计算、基于P2P的点对点计算、基于服务器集群的“云计算”等。虽然方法众多，但却需要根据实际情况进行选择，以Google为代表的“云计算”以其应用简单、高效得到了广泛认可。它通过在分布式文件系统GFS基础之上的MapReduce编程模型以及廉价集群的建立，解决了许多大规模数据的计算问题。

在众多科技合作网络的分析中，社团划分以及网络演化分析是人们比较关注的两个方面。传统的社团划分方法在处理大规模数据集时，其性能总是很难满足要求，甚至当数据集过大时，单机内存根本无法承受而造成算法无法执行。所以本文提出了一种建立在云计算基础上的文献数据的社团划分方法，此方法在处理大规模数据集时是有效的。同时，我们还在社团划分的基础上对合作网络的演化进行了分析。通过分析，可以清晰地看到活跃在学术会议中的科研合作团队及团队的核心科研工作者，这对进一步了解学术会议研究方向及动态具有指导性作用。本文将按如下结构组织：首先介绍基于云计算的图挖

* 本文得到国家自然科学基金项目（90924029）和国家“十一五”科技支撑计划项目（2006BAH03B05）资助。

掘与文献网络演化分析方面的相关工作；接着将介绍基于云计算的文献数据的社团划分方法；然后通过实验对比一下性能；最后将在社团划分的基础上分析科研合作网络的演化。

2 相关工作

在图挖掘和网络分析的相关应用中，极大团挖掘和社团发现扮演着重要的角色。这不仅因为其复杂性[1]，还因为来自工程应用的需求[2]。此外，针对这些问题，除算法优化和局部改进外，也有些研究工作使用了分布式算法，如[3,4]等。然而，这些方法只关注理论上的并行化，并没有提供一个系统化的方法。作为目前分布式计算的一个重要基础计算平台，MapReduce受到了广泛的关注。在研究领域，尤其是知识发现领域，MapReduce在海量数据处理方面扮演着越来越重要的角色。Chu等人[5]在MapReduce平台上实现了多种Machine Learning的算法，并用KDD Cup 99等数据集进行相关的实验对比；Papadimitriou等人[6]提出了一种分布式聚类框架，并在Hadoop上应用TREC等数据集进行了大规模数据的聚类实验，聚类的性能得到了很好的提升。Tang等人[7]在云计算平台上实现了文献数据的课题影响性分析。

网络演化分析作为一个新兴的领域，近年来受到了广泛的关注。从研究角度来讲，Jin等人[8]针对增长型的网络结构，尤其是WWW，提出了两种有效的描述模型。Leskovec等人[9]则应用arXiv等数据从网络的基本特征入手（如边点比、平均最短路径等），揭示了网络在演化过程中呈现出的与静态网络所不同的特性。Backstrom等人[10]通过应用DBLP等数据集研究动态网络中社团的演化特征，发现个体加入社团与社团结构有着密切的关联。Tantipathananandh等人[11]提出了一种基于染色方法的算法框架来进行社团演化分析。虽然这些研究者对网络演化较早地进行了关注，并做了大量开创性的工作，但其方法都存在着不足，如认为网络的演化只是一个不断增长的过程，忽略了对网络随机性和突发性的考虑等。最近，一些研究者开始注意到传统方法的不足。其中与本文关联密切的研究有：Leskovec等人[12]基于MLE（Maximum likelihood estimation，最大相似性估值）方法从多个维度考察了MSN网络的组织特性。Tong等人[13]提出了Colibri方法集来处理静态和动态的网络分析。Lin等人[14]提出了FacetNet框架，整合了社团发现和演化的分析方法。为了避免演化带来的噪声，该框架中的社团发现算法不仅依赖于当前的网络结构，还考虑了过去的网络特征。Asur等人[15]认识到网络演化中事件发生的必然性，提出了分析个体和社团演化行为特征的框架并且应用DBLP等数据集进行系统相关的测试。在该框架中，关键事件被用来预测社团的发展趋势。在Sun等人[16]提出的GraphScope框架中，二分图被用来描述整个网络，并用来进行社团划分。

3 基于云计算的科研实体网络的社团发现算法

在传统的图挖掘实现中，图往往被表示成邻接链表的形式并被加载到内存中，接下来的算法可以通过索引的方式访问到所有节点进而实现其算法目的。这种简单直接的实现方式存在两种缺点：1）随着图规模的显著扩大，图并不能够完整加载到内存中；2）基于单内存单机的图挖掘算法不满足性能上的需求。为了应对这两方面的挑战，本章提出一种分布式的社团发现流程，其过程包括极大团发现（K-Enumer）、子连通分量合并（C-Merge）到最终的社团结果。

3.1 极大团发现（K-Enum）

图的邻接链表表示形式能够表达图中一跳（one-leap）信息，即节点v及其邻居Γ(v)。然而，为了获得极大团，需要一种表现形式，其至少能够表达图中两跳（two-leap）的信息，即节点v、邻居Γ(v)和邻居的邻居Γ(Γ(v))。

例如图1中的图，为了得到节点1所涉及的极大团，在图的邻接链表中需要获得< 1; <2; 3; 4 >>、< 2; < 3; 4 >> 和 < 3; 4 >。在实际中，尤其是单机算法中，只需要简单的索引技术便可获得这3条记录。而在分布式计算环境中，输入通常被分配到不同的计算节点进行计算，而且为了保证效率，这些节点之间往往不进行通信。这样，相关的记录就有可能被分配到不同的计算节点，进而较难进行极大团的发现。

为了把相关的记录有效地组织

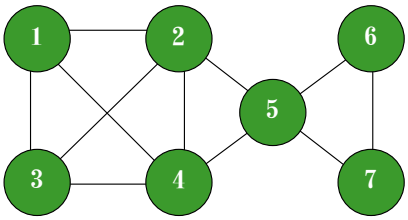


图1 图示例

在一起，使节点的两跳信息能够被分布到同一个计算节点上，这里提出一种两跳（two-leap）的结构转化形式。给定一条记录< 1; < 2; 3; 4 >>，其表示了节点1和其邻居< 2; 3; 4 >，将其转化成如下形式：

< 2; < 1; < 3; 4 >>>

< 3; < 1; < 2; 4 >>>

< 4; < 1; < 2; 3 >>>

该转化可以在Mapper中完成。对邻接链表中的其他记录，进行相同的转化便可转化成两条形式。这样，给定一个图，经过这步转化的结果记录就包括了每个节点和其所有“两跳”记录中的一条（如上例）。所以如果能够收集到该节点和其所有两跳记录，那么就可以构造以该节点为顶点的一棵子树（或子图）。例如，给定一节点4，依图1，收集到其所有两跳信息如下：

< 4; < 1; < 2; 3 >>>

< 4; < 2; < 3; 5 >>>

< 4; < 3; < 5 >>>

基于这些记录，可以构造出以4为顶点的一个子树，然后基于顶点4做深度优先搜索，即可获得4所在的极大团(1; 2; 3; 4)。

3.2 社团发现

显然，极大团在结构上的强约束使其实际应用受到限制，而社团结构传递了更多的信息。因此，许多研究关注在社团发现算法和用社团结构来解释复杂网络中的组织结构或社会性特征。然而，过去的研究工作往往受到社团发现算法的复杂度及网络规模的限制。因此，本节将提出一种分布式社团发现算法。

基于上节挖掘出的极大团，K-Merge（Key-based clique merge）对挖掘出的极大团进行初始合并。由于极大团挖掘过程中采用了顶点优先策略，所以在上节结果中存在大量具有相同顶点的极大团，这些极大团如果满足如下阈值即可以进行初始合并。

$$V(C_1) \cap V(C_2) = \min(V(C_1), V(C_1)) - 1 \quad (1)$$

和[17]相比，这个步骤不需要参数并且更加严格，从而保证了算法结果的合理性。

经过初始合并后，接下来构造一个社团之间的关系图，叫做CK-Graph。在这个图中，每个节点代表一个社团，而边代表社团之间的关联关系，即两个社团满足公式(1)。这步构造过程仍然可以用MapReduce进行。

自然，下一步将基于该社团关系图进行连通分量的发现。在单机中，该算法只需要使用图搜索算法即可完成，然而在分布式计算中，该算法并不能一步完成。这里提出一种迭代式的连通分量挖掘方法，即N-Merge和NN-Merge。这两步的输入是图的两跳信息，所以首先需要使用G-Tran进行图转化，然后N-Merge负责合并邻居记录，NN-Merge负责合并非邻居记录。其中N-Merge算法描述如下：

- mapper1输入的是两跳表示的图结构数据：其中key是节点，value是节点的两跳内的邻居节点。对每一条记录内的节点为key，该记录为value输出到reducer1。
- reducer1收集到每一节点为key的相关记录后，去掉重复的边，这样就得到了结果。

NN-Merge和N-Merge仅有的不同在于Mapper1，即在N-Merge中，Mapper1的输出是<v', v>，其中$v' \in \Gamma(v)$。而在NN-Merge中，$v' \in \Gamma(v)$。循环运行N-Merge和NN-Merge直至没有合并再次发生，其输出结果即为社团关系图中的所有连通分量，也即最后的社团结构（把相连通的社团合并在一起）。

4 对比试验及结果分析

4.1 硬件环境

为了运行分布式算法并处理海量数据，这里架构了一个包含32个节点的集群系统。其中每个节点的配置为Intel Xeon 3.20GHz*2, 2GB RAM和6T的总存储容量。在这个集群系统上运行了Linux RH4操作系统并部署了Hadoop计算平台。

4.2 数据及基本实验说明

为了说明本文提出的分布式算法的高效性，这里使用的数据集是SCI和DBLP。这两个数据集都来自Newman的公开科研合作网数据集[①]，其中SCI数据集包括39.2万个节点以及87.4万条边，DBLP数据集包含49万个节点和241.6万条边。

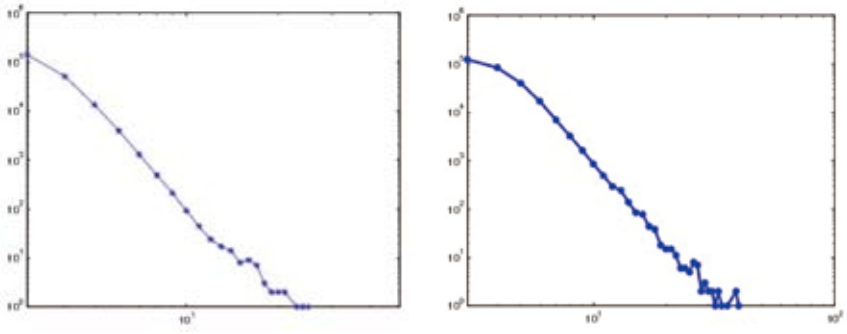
为了进行对比，这里实现了两个广泛使用的极大团挖掘算法，[18]（简称为BK）和[19]（简称为TTT）。作为实验参考，首先针对每个数据集在同一个计算节点上分别运行BK和TTT算法，为了避免时间波动，每个算法运行5次，实验结果取平均值（见表1）。表1中还列出了抽取出来的极大团的基本统计并在图1中展示了极大团的分布特征。

① 此数据集来自http://www-personal.umich.edu/ mejn/netdata/。

表1 基本统计

Name	|E|/|V|	K_{max}	N_C	N_{c-3}	Max_c	BK(s)	TTT(s)
SCI	2.23	496	213K	14K	25(1)	13	21
DBLP	4.93	1119	281K	12K	40(1)	25	35

注：|E|/|V|为图的边点比；K_{max}为点的最大度；N_C为图中极大团的个数；$N_{c=3}$为3极大团的个数；Max_c为最大极大团的大小及个数；BK(s)为使用BK算法的运行结果，单位秒；TTT(s)为使用TTT算法的运行结果，单位秒；“#”表示使用该算法无法在有效时间内（<3600s）运行完成。



(a)SCI (b)DBLP

图2 极大团分布

4.3 图转化及极大团发现

MapReduce计算平台的一个显著特性在于能够透明灵活地提供给不同计算任务不同的计算资源。在本节实验中，图转化操作的性能与计算节点的个数几乎为正比关系。这是因为除I/O操作外，图转化是一个轻量级的操作（非CPU bound）。

表2展示了分布式算法在各个数据集上的性能实验。每个实验数据在不同的Reducer数目（用R表示）下运行多次并取平均值。本实验中的Mapper数使用了系统默认值，即M=S/64MB，其中M为输入数据的大小，64MB为Hadoop默认的分块大小。从表2可以看出，对于前两个数据集，即使最快的分布式运行时间也要比单机版本慢。此外，针对不同的Reducer配置数目，实验的性能差别并不明显。这两个实验表明，针对较小的数据集，由于有额外开销，例如任务调度、网络传输、结果的排序（MapReduce平台提供）以及与DFS的交换等，分布式的计算并不能带来明显的性能提升。因此，MapReduce尤其适用于大规模的数据处理。

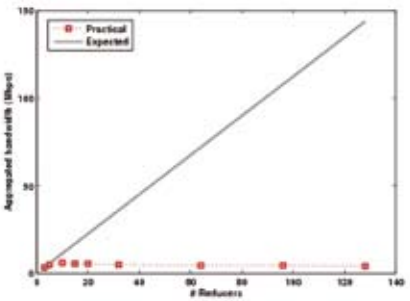
表2 分布式极大团挖掘性能实验（单位：秒）

Name	R=3	R=5	R=10	R=15	R=20	R=32	R=64	R=96	R=128	BM
SCI	35	24	20	22	22	24	27	27	28	13
DBLP	140	90	49	34	30	33	35	35	38	25

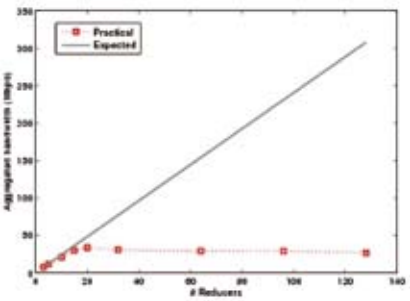
注：BM表示Benchmark，其值来自于表4-1中BK和TTT算法时间的较小者。

表2展示了在大规模数据集上应用分布式计算的效率。和小数据集相比，其分布式性能得到了充分的体现。

与单机进行直接的时间比较并不能说明分布式算法的执行效率问题。所以这里转化为通过累积吞吐率来对算法的并行效率进行评价。从理论上讲，如果单机算法的吞吐率为T1的话，那么具有n个Reducer配置的并行算法吞吐率应该为n*T1。图3展示了实验效率和理论效率的对比，其中蓝色为理论值，灰色为实际实验结果。从图中可以看出，每个数据集的实际实验结果都经历了一个“增长—平稳”的过程。在初始时，即Reducer的



(a)SCI



(b)DBLP

图3 基于累积吞吐率的性能比较

数目小于20时，实验结果和理论结果几乎一致。也就是说，在这个阶段，可以通过简单地增加Reducer数目而达到实验效率的倍增。同时还可以看出，效率最快的提升往往发生在Reducer的数目为32到64的时候，这时恰好是计算平台的硬件机器数（32个计算节点）和CPU核数（32*2）。在这个最优效率提升后，算法的效率提升将变得缓慢，并与理论值相差开始加大。这也表明，在最优的Reducer的数目配置后，如果再简单地增加Reducer数目并不能获得相应的效率提升。

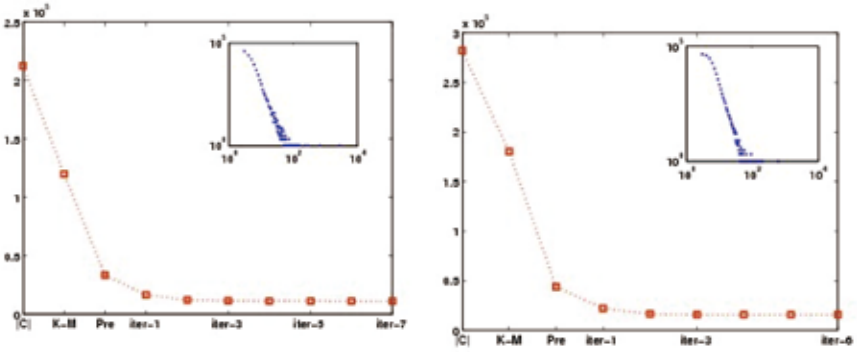
4.4 社团发现

本小节使用上一节的极大团挖掘结果作为输入来进行分布式的社团挖掘，同时希望通过对真实环境中的大规模图结构进行挖掘进而发现一些有意义的社团结构。表3总结了各步骤的实验结果，可以发现，虽然输入的极大团数据大小不一，从0.2百万到8百万（将近40倍），但每个步骤的时间开销却差别不大（最大值仅为最小值的2到3倍）。也就是说，这里使用的算法并不随数据的大小成线性增加。这种性质尤其适合于超大规模数据。

图4展示了经过每轮合并后，图中社团的数目。该迭代步骤一直执行到社团数目不再变化。其中每个图中的子图表明了最终社团的分布情况。“iter-n”表示运行了N-Merge和NN-Merge第n次后的社团数据。从图4可以看出，为了获得最终的合并结果，迭代步骤需要运行6到9次。然而在实际中，当经过大约4次迭代后，社团数目的变化就会很小了（小于1%），并且这时的结果和最终结果的误差率也在1%以下。所以在实际使用时可以在效率和精度上作出均衡。

表3 社团发现

Name	K-Merge	CG	Pre	Avg. iter	Nc
SCI	20s	44s	19s	21s	103K
DBLP	22s	46s	20s	25s	166K



(a) SCI　　(b) DBLP

图4 社团发现过程（内坐标为社团分布）

5 学术会议的社团演化分析

我们将以数据挖掘领域内的KDD学术会议为例，跟踪其近五年内的科研合作团队的演化情况，进而为掌握每年学术会议的进展情况提供指导。图5~图9展示了KDD学术会议从2004年到2008年科研合作团队的变化情况。特别地，科研团队划分使用的算法为上面描述的算法，用红色实心圆点标识出了科研团队的核心成员。与此同时，在图的下文标识出了科研合作团队中节点编号所对应的科研工作者的名字。

从图5中，我们可以看到在2004年，有五个显著的科研合作团队，且形成了三个大的连通分量。特别地，2668号节点代表着数据挖掘领域的专家韩家炜，而与其所在科研合作团队相连的科研合作团队中的5279号节点则为其学生，而10884也是IBM著名的数据挖掘方向的专家。而125308号节点为Christos，其为图挖掘方向的专家，而2664节点为卡耐基梅隆大学Andrew W.Moore教授，其研究方向有数据挖掘、机器学习等，则代表着一个独立的小团队。

到2005年时，KDD中所形成的科研团队如图6所示，其显著的特征就是：特别分散，不存在较大的科研团队及连通分量，而存在大量的小社团，在此我们仅列出了规模在前四名的科研合作团队。此时我们依然可以看到2668、2464节点，这说明他们连续两年都活跃在KDD学术会议中，而此时又出现了两个新生的科研团队，分别以59260、33744节点为核心节点，经核实我们发现Ravi Kumar（59260）为Yahoo研究院Web Mining的专家，

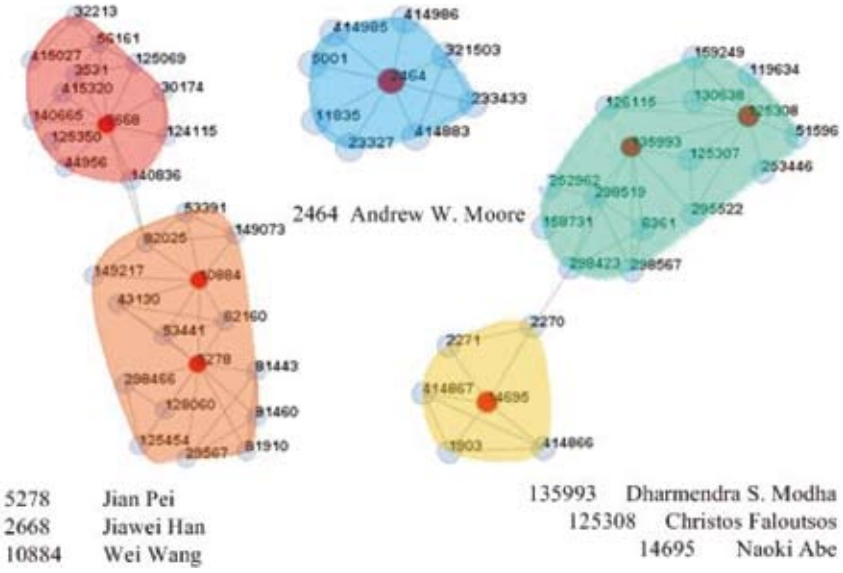


图5 2004年KDD科研合作团队

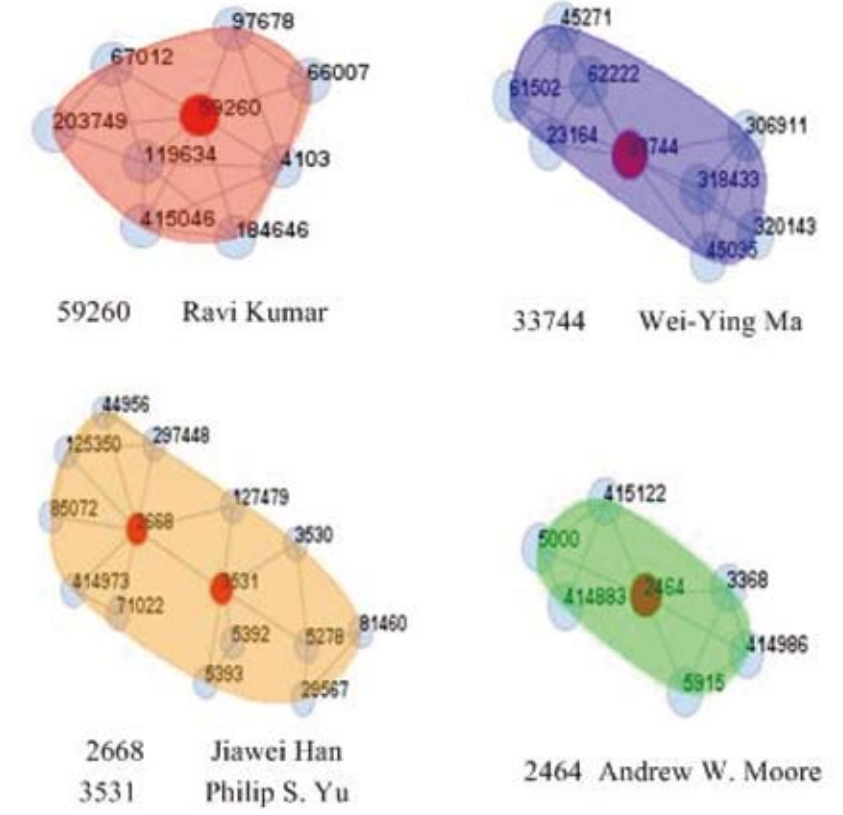


图6 2005年KDD科研合作团队

主要从事Web Search方面的工作。而Wei-Ying Ma（33744）则为数据挖掘及自然语言处理方向的专家。

而在2006年，形成的科研合作社团（如图7所示）规模明显比2004年与2005年的大，同时出现了以节点7096为核心的新的科研合作团队。而节点3531的加入使得在2004年以10884和5278为核心的社团规模变得更大。

在2007年，又形成了大量新生的科研合作团队，特别地，相比于2006年的科研合作团队，核心10884、3531在2007年分别代表着不同科研合作团队（如图8所示）。

在2008年的科研合作团队中（如图9中所示），最显著的特点是：以2668、3531、5278、125308为核心的科研合作团队在此时形成了一个大的连通分量，这也说明在数据挖掘领域内各不同的研究方向内的科研合作专家间的合作也越来越多，而科研合作的范围也在变得越来越广。

综上所述，本节呈现出了KDD近五年中的科研合作网络中所形成的科研合作团队，从中可以清晰地看到活跃在学术会议中的科研合作团队及团队的核心科研工作者，这对进一步了解学术会议研究方向及动态具有指导性作用。但本文只是提供了科研团队的概况，如何进一步确定其科研方向，仍需要做进一步的研究。

6 结论与展望

海量的文献数据提供了丰富的科技信息挖掘分析资源。区别于传统科学计算学，图挖掘可以对文献数据进行多方面的分析，比如社团发现、社会影响性分析、合作网络演化分析等。这些分析有助于获知各个领域的发展动态、资源分布、交叉关联关系以及科研团队之间的合作关系等信息，以便很好地理清科技发展的脉络，合理地分配

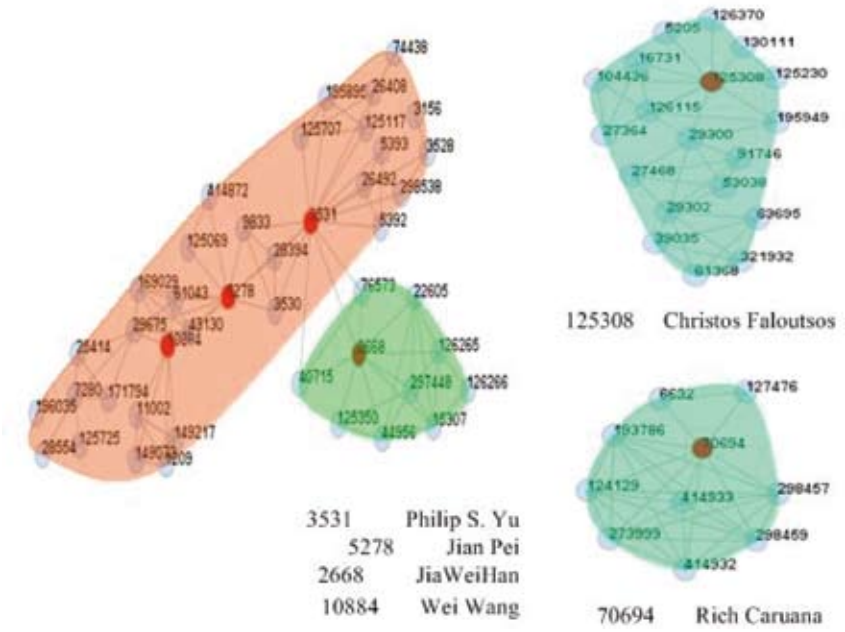


图7 2006年KDD科研合作团队

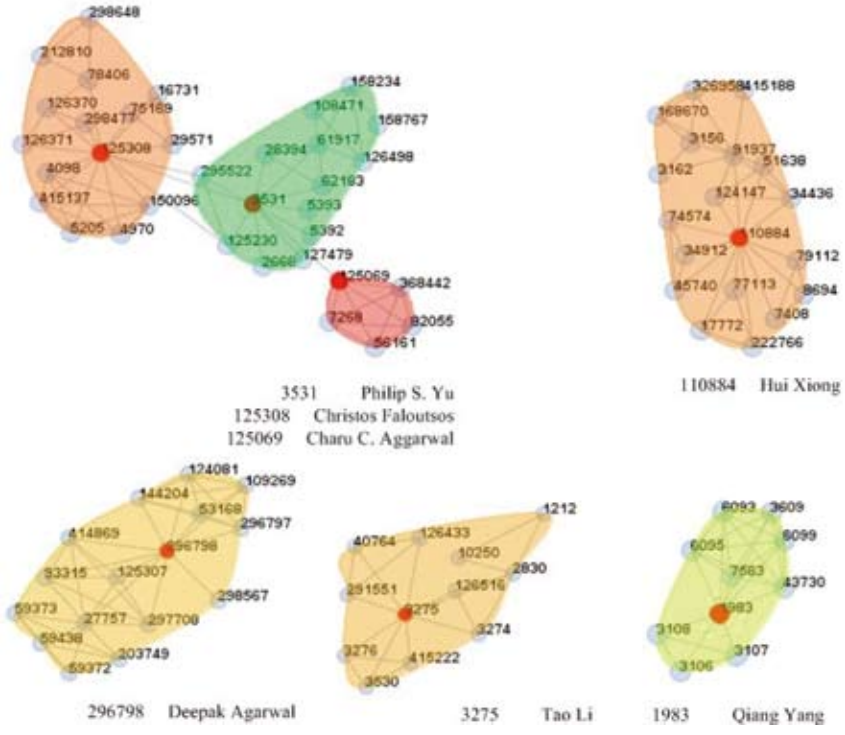


图8 2007年KDD科研合作团队

各种资源，同时可以促进科技走向市场化。因此，如何快速有效地对大规模文献数据进行分析是需要解决的问题。然而，由于图挖掘的特殊性，图挖掘算法在应用到云计算平台上时有一定的限制性，也就是说，并不是所有的图挖掘算法都适合在云计算平台上实现。所以，找出适合于云计算平台并能够解决问题的算法是当前大规模图挖掘的目标。本文在云计算平台上实现了对大规模文献数据的社团挖掘算法，并在SCI和DBLP两个数据集上实验了该算法，通过实验的性能对比，证明该基于云计算平台的算法应用于大规模问题是高效的。同时，本文给出一个分析实例，利用该算法对KDD会议作者合作数据集进行了社团划分，并对社团划分的结果进行了演化分析，并找到了活跃在学术会议中的科研合作团队及团队的核心科研工作者。而如何把云计算平台应用集成到文献数据挖掘系统中，对科技文献数据进行多角度、高效率地分析将是我们下一步的工作。

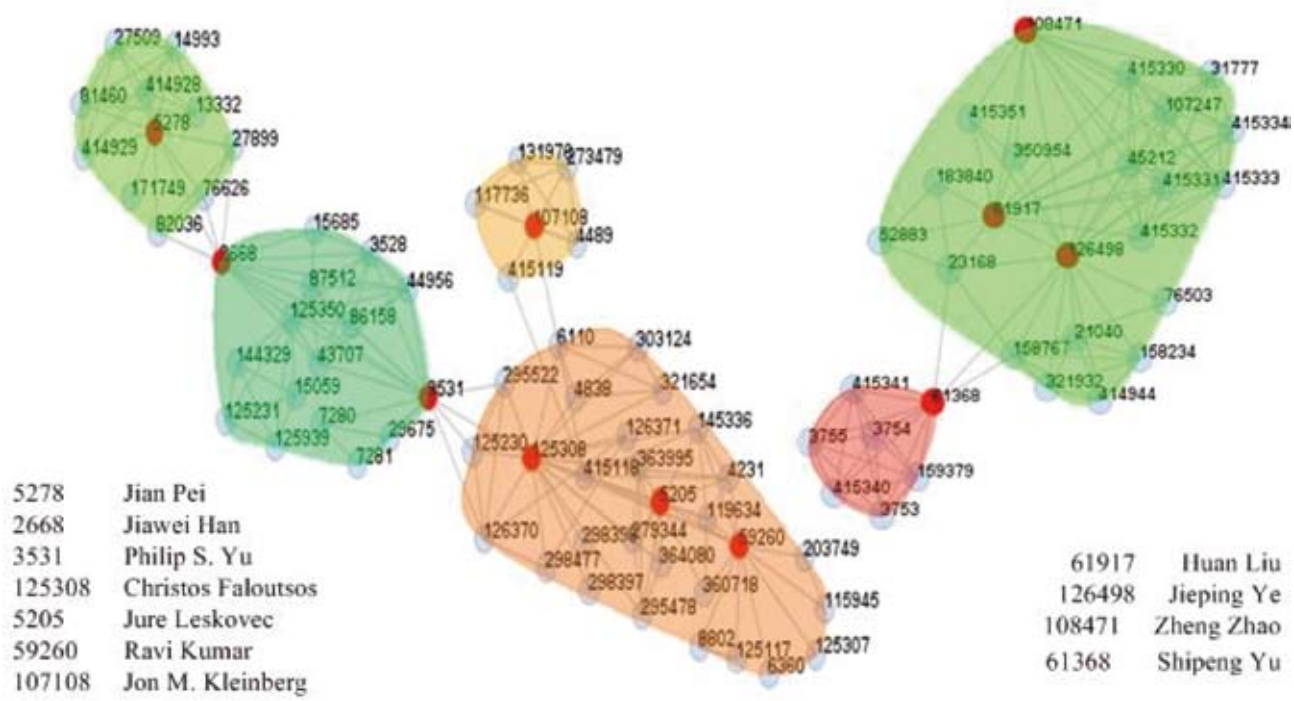


图9 2008年KDD科研合作团队

参考文献

[1] JOHNSON D S, PAPADIMITRIOU C H. On generating all maximal independent sets [J]. Info. Proc. Lett., 1988,27(3):119-123.
[2] PALLA G, DERENYI I, FARKAS I, VICSEK T. Uncovering the overlapping community structure of complex networks in nature and society [J]. Nature, 2005,435(7043):814-818.
[3] DU N, WU B, XU L, WANG B, PEI X. A parallel algorithm for enumerating all maximal cliques in complex network [C]// Proceedings of the Sixth IEEE International Conference on Data Mining – Workshops, 2006:320-324.
[4] KARP R M, WIGDERSON A. A fast parallel algorithm for the maximal independent set problem [J]. J. ACM., 1985,32(4):762-773.
[5] CHU C T, KIM S K, LIN Y A, YU Y, BRADSKI G R, NG A Y, OLUKOTUN K. Map-reduce for machine learning on multicore [C]// NIPS ’ 06, 2006:281-288.
[6] PAPADIMITRIOU S, SUN J. Disco: Distributed coclustering with map-reduce (a case study towards petabytescale end-to-end mining) [C]// ICDM'08, 2008:512-521.
[7] TANG J, SUN J, WANG C, YANG Z. Social Influence Analysis in Large-scale Networks [C]// KDD '09, 2009:807-815.
[8] JIN E M, GIRVAN M, NEWMAN M E J. Structure of growing social networks [J]. Physical Review E., 2001,64(4):046132.
[9] LESKOVEC J, KLEINBERG J, FALOUTSOS C. Graphs over time: densification laws, shrinking diameters and possible explanations [C]// KDD '05, 2005:177-187.
[10] BACKSTROM L, HUTTENLOCHER D, KLEINBERG J, LAN X. Group formation in large social networks: membership, growth, and evolution [C]// KDD '06, 2006:44-54.
[11] TANTIPATHANANANDH C, BERGER-WOLF T, KEMPE D. A framework for community identification in dynamic social networks [C]// KDD’ 07, 2007:717-726.
[12] LESKOVEC J, BACKSTROM L, KUMAR R, TOMKINS A. Microscopic evolution of social networks [C]// KDD '08, 2008:462-470.
[13] TONG H, PAPADIMITRIOU S, SUN J, YU P S, FALOUTSOS C. Colibri: fast mining of large static and dynamic graphs [C]// KDD '08, 2008:686-694.
[14] LIN YR, CHI Y, ZHU SH, SUNDARAM H, TSENG B L. Facetnet: a framework for analyzing communities and their evolutions in dynamic networks [C]// WWW '08, 2008:685-694.
[15] ASUR S, PARTHASARATHY S, UCAR D. An event-based framework for characterizing the evolutionary behavior of interaction graphs [C]// KDD '07, 2007:913-921.
[16] SUN J, FALOUTSOS C, PAPADIMITRIOU S, YU P S. GraphScope: parameter-free mining of large time-evolving graphs [C]// KDD '07, 2007:687-696.
[17] PALLA G, DERENYI I, FARKAS I, VICSEK T. Uncovering the overlapping community structure of complex networks in nature and society [J]. Nature, 2005,435(7043):814-818.
[18] BRON C, KERBOSCH J. Algorithm 457: finding all cliques of an undirected graph [J]. Commun. ACM, 1973,16(9):575-577.
[19] TOMITA E, TANAKA A, TAKAHASHI H. The worst-case time complexity for generating all maximal cliques and computational experiments [J]. Theoretical

Computer Science, 2006,363(1):28-42.
[20] NEWMAN M E J. The Structure and Function of Complex Networks [J]. SIAM REVIEW, 45(2):167-256.
[21] WATTS D J, STROGATZ S H. Collective dynamics of ‘small-world’ networks [J]. Nature, 1998,393(6684):409-410.
[22] BARABASI A-L, ALBERT R. Emergence of Scaling in Random Networks [J]. Science, 1999,286(5439):509-512.
[23] GRIVAN M, NEWMAN M E J. Community structure in social and biological networks [J]. PNAS. 2002,99(12):7821-7826.
[24] TORRELLAS J. Architectures for Extreme-Scale Computing [J]. Computer, 2009,42(11):28-35.
[25] DEAN J, GHEMAWAT S. Mapreduce: Simplified data processing on large clusters [C]// OSDI ’ 04, 2004:137-150.
[26] COHEN J. Graph twiddling in a mapreduce world [J]. Computing in Science and Engineering, 2009,11(4):29-41.
[27] KANG U, TSOURAKAKIS C E., AND FALOUTSOS C. PEGASUS: A Peta-Scale Graph Mining System – Implementation and Observations [C]// ICDM ’ 09, 2009.

作者简介

杜雅红，硕士研究生，主要研究领域为数据挖掘以及复杂网络。通讯地址：北京邮电大学179信箱 100876。E-mail: du_yahong@126.com
吴斌，副教授，主要研究领域为数据挖掘、复杂网络及智能信息处理。通讯地址：同上。E-mail: wubin@bupt.edu.cn

Community Detection and Evolution Analysis of Co-authorship Networks Based on Cloud Computing

Du Yahong, Bai Yunlong, Wu Bin / Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia, Beijing University of Posts and Telecommunications, Beijing, 100876

Abstract: With the growing amount of bibliographies and expanding of scientific research, more in-depth research about bibliographic data is needed. At the same time, the continued exponential growth in both the volume and the complexity of information is giving birth to a new challenge to data analysts. To meet this challenge, a new class of techniques and computing platforms, such as MapReduce model, which mainly focuses on scalability and parallelism, has been emerging in research and industry area. An innovative community detection algorithm based on Mapreduce is proposed in this paper. Detailed analysis of a dataset of one academic conference is given as a case study of community evolution.

Keywords: Bibliographic data, Graph mining, Community detection, Community evolution, Cloud computing

（收稿日期：2010-05-31）