浙江大学

# 本科生毕业论文（设计）
# 开题报告


学生姓名：周晓龙

学生学号：3071102314

指导教师：何钦铭

年级与专业：计算机科学与技术2007级

所在学院：计算机学院

一、题目： 动态社会网络团体发现与分析算法研究

二、指导教师对开题报告、外文翻译和文献综述的具体要求：

开题报告：

1. 介绍动态社会网络团体挖掘的背景。

2. 分析现有相关技术。

3. 明确研究的内容和任务。

4. 提出切实可行的研究方案和计划。

文献综述：

1. 阅读一定数量的文献。

2. 总结现有动态社会网络团体挖掘的常用算法。

3. 分析相关算法的优缺点。

外文翻译：

翻译一篇关于动态社会网络团体挖掘的论文的核心思想和算法，要求翻译准确，语句通顺，结构完整。

指导教师（签名）_____

年　　月　　日

# 毕业论文（设计）开题报告、外文翻译和文献综述考核

答辩小组对开题报告、外文翻译和文献综述评语及成绩评定：

| 成绩比例 | 开题报告<br>占（20%） | 外文翻译<br>占（10%） | 文献综述<br>占（10%） |
|---|---|---|---|
| 分 值 | | | |

开题报告答辩小组负责人（签名）_____

年　　月　　日

# 目 录

**本科毕业论文（设计）开题报告**

# 动态社会网络团体发现与分析算法研究

# An Approach to Discover and Analyze Communities in Dynamic Social Networks

## 1. 课题背景

社会中人与人之间的错综复杂的关系，可以抽象成一个非常复杂的网络图，每个人就是这个图中的一个节点，而人与人之间的关系就是这个图的边。人和人的关系有陌生有紧密，紧密的关系（如共同的兴趣、共同参与某类事件）使相关人们形成一个团体。研究社会网络中的团体挖掘算法、团伙的进化规律对社会学研究和相关应用有非常重要的意义。例如，对犯罪分子的犯罪网络团体的挖掘和研究能够帮助公安机关深层次地挖掘埋藏在人际关系中的重要线索，辅助公安刑侦人员的案件侦破。

通常来说，随着时间的变化，社会网络中的团体构成随时都在发生着变化。相关的团体可能发生如分裂、合并、生长、消亡等变化。研究动态的社会网络中团体的变化规律，有很好的应用前景，如帮助社会学家研究社会发展的相关规律，帮助商家和受众掌握最新的流行趋势，帮助政府部门根据社会的发展制定最优决策；研究互联网社区(如微薄、论坛等)团体的进化，有助于相关组织者把握最新的流行趋势，更好地服务参与者(如微薄用户、电子商务消费者等)，也能对商业

决策提供有力的支持等。

将社会中人与人之间的联系表现为人们的电话通信记录、邮件记录、共同参加一个活动的记录等等，使用一定的算法和数学模型对这些数据进行建模，可以把社会网络抽象为一个网络图。如 Figure 1 所示的就是一个博客社区的关系网络图，每个点代表一个博客，边代表博客文章的引用和评论关系，这个网络的结构随着博客社区的演化发生演化。

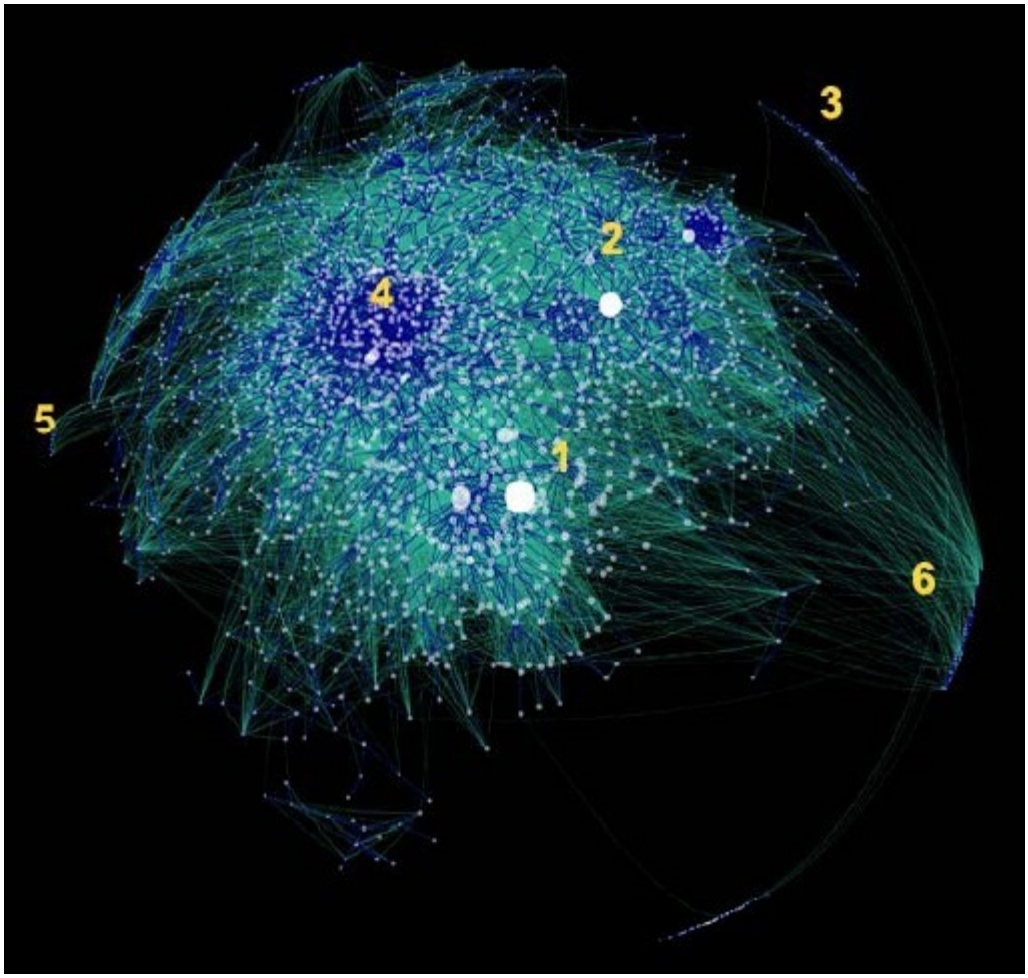


Figure 1：Blogosphere 博客圈数据网络图。社会网络图的一个示例

研究基于这些网络图上的团体发现和分析的相关算法，并开发有关的计算机程序，可以自动化地高效地分析数据，挖掘出有用的信息和规律。如 Figure 2 所示的是一个研究者合作研究的社会网络，节点代表科学家，边代表他们合作参与

共同研究的关系。通过一些社会网络挖掘的算法，可以将具有共同兴趣趋向的科学家聚集成团体，图中用不同的颜色代表不同的团体。

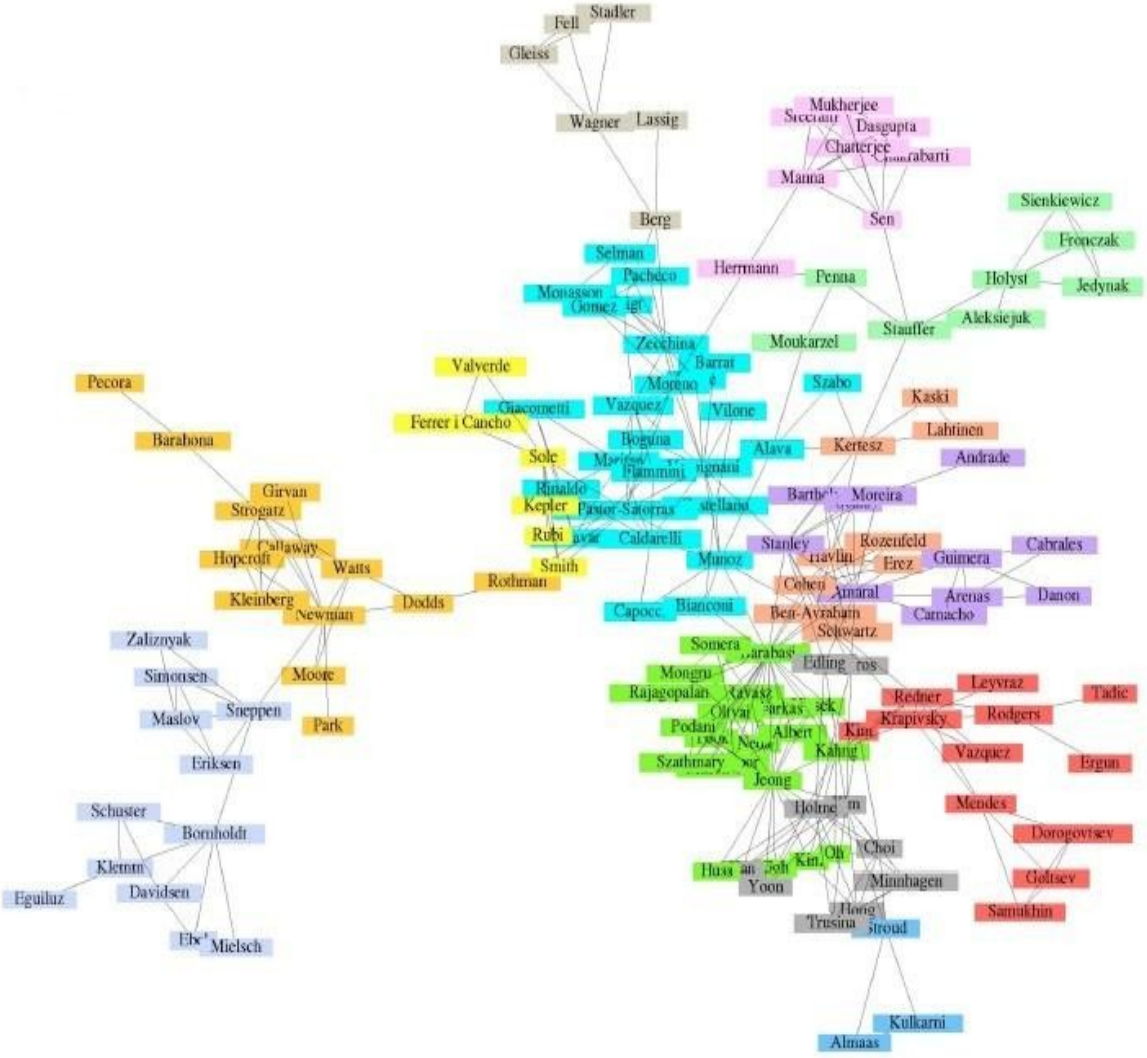


Figure 2：研究合作作者关系网络，社会网络的一个示例

# 2. 目标和任务

对于静态的网络进行团体发现和挖掘，已经有相当多的研究和算法，如层次聚类、k-means 聚类、betweenness 切片划分算法[9]等等。这些算法能够根据

不同的聚类相似度依据对静态网络进行分析并挖掘出其中的团伙。对于静态网络，也有很多的研究和方法来分析团伙内成员、团伙间的联系。关于静态网络的数据挖掘现在已经有非常成熟的知识体系和方法。但是，现有的静态网络的挖掘方法对实际分析的支持还并不充分，因为实际的社会网络，总是处在时刻变化中。如果有一套行之有效的对于动态网络的分析挖掘方法，将为相关领域提供非常大的帮助。

与静态网络不同，在动态网络中，由于节点（人）之间的联系随着时间轴的推进而不断变化，而网络结构也时刻在发生着改变。使用传统的静态分析挖掘方法就难以提供有效地对动态网络进行分析和挖掘。本次研究的目的，就是研究出一套较有效的挖掘方法和算法对动态网络进行分析，挖掘出动态网络中的团伙信息，并应用这套方法分析和研究在动态网络中的团伙进化现象及进化规律。概括之，其主要研究点包括以下几点：

1) 研究并得出一套有效的对动态网络进行挖掘和分析的方法和算法
2) 设计软件实现相关的算法
3) 使用实际数据集对实现的算法进行实验验证，对实验结果进行分析。针对数据集探讨动态网络中团伙的演化现象和演化规律

# 3. 研究方案和关键技术考虑

对于基于时间线的动态社会网络，可以定性为分析在以时间维度为基线，网络中团体的演变过程和个人的活动过程。以时间维度为基线，一个最行之有效的方法就是对动态网络在各个时间点上的状态做切片，然后分析各切片之间的演化。

从这个思路出发，本次研究的两个研究重点为：1.单时间片上的静态团体挖掘；2.时间片之间的团体演化分析与个人活动分析。

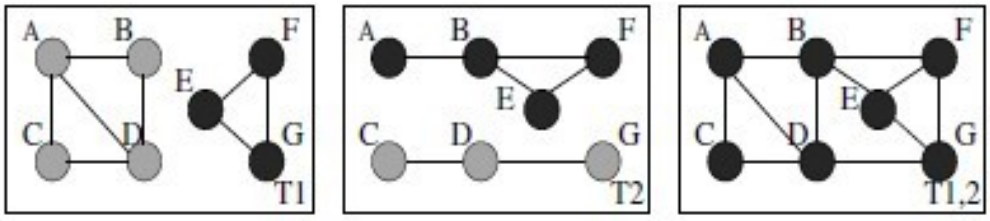


Figure 3：时间切片的概念，从左到右分别是：a)时间切片 t=1；b)时间切片 t=2；c)时间累积切片 t=1,2. 这是 t1 时间片和 t2 时间片的关系叠加

所谓切片，并不是时刻点上的横断面，它是一个小时间区间Δt 上的网络关系聚合。例如对于基于通话记录的网络图切片，切片 t1 是用户 1~3 月的通话记录建模构建的网络图，切片 t2 是用户 4~6 月的通话记录建模构建的网络图。t1 与 t2 的叠加即为根据用户 1~6 月通话记录建模后构建的网络图。Figure 3 所示就是 3 个切片的示意图。T1 是在时刻 t=1 时的网络图结构;T2 是 t=2 时刻的结构。T1,2 是 T1 时间片和 T2 时间片的聚合。

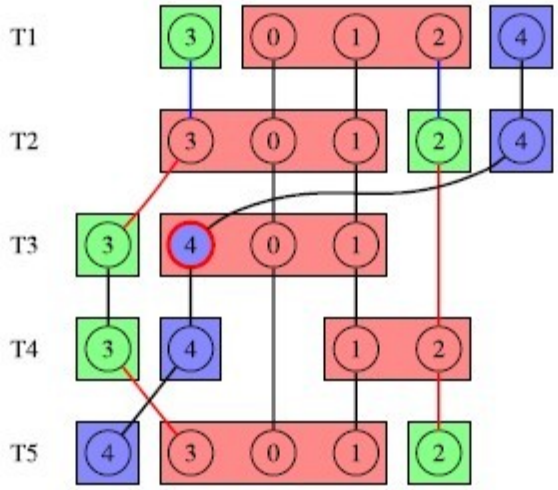


Figure 4：图中 Ti 是时间片序列。随着时间的变化，图的结构发生着变化。展示了 5 个时间片上不同的图结构，以及团伙的保持和进化

随着时间的推移，网络图时刻在发生着演化。如 Figure 4 所示，T1~T5 五个

时间片上 ,网络的结构各不相同。随着时间的推移 ,网络中的团伙也在发生演化，图中用相同颜色来标注相同的团伙。

## 3.1. 单个时间片上的团伙聚集和分析

静态的网络挖掘，我们采用一个聚类分析的过程进行团体挖掘。将采用的方法是 Michelle Girvan 和 Mark Newman 提出的 Betweenness 切边算法[1]。算法基本思路比较简单。统计团伙中两两节点之间的最短路径，边上经过的最短路径条数称作 betweenness。Betweenness 越大的边上通过的最短路径条数越多。连接团伙间的边承载了团伙间的最短路径流，其 betweenness 值比较大，可以称为关键路径。通过去掉网络中的关键路径，即可以发现团伙。

在 Betweenness 切边算法中，其算法复杂度的关键在于发现关键路径的计算上。经典算法中根据 betweenness 的定义计算图中两两节点间的最短路径并统计每条边的 betweenness 值。通常 ,Betweenness 计算的时间复杂度为 $O(n^3)$，空间复杂度为 $O(n^2)$，其中 n 为网络中节点的个数。我将采用 Ulrik Brandes 提出的一个优化的 betweenness 算法[8]，其计算空间复杂度为 O(m+n)，时间复杂度为无权图 O(mn)或有权图 $O(mn+n^2\log n)$，其中 n 为网络中节点的个数，m 为网络中边的条数。

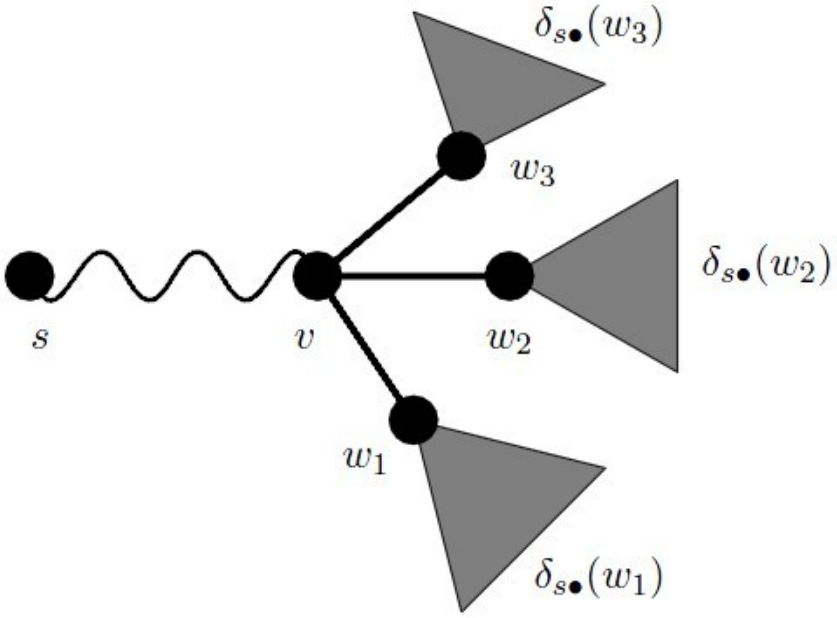


Figure 5：改进的 betweenness 算法示意图

在改进的 betweenness 算法中，Ulrik 引入节点权重记录最短路径流，运用动态规划的思想，利用已经计算完成的路径的关键度权重来支持后面的计算，减少了重复计算，提高算法效率。如 Figure 5 所示，三个阴影的三角区域是已经计算好 betweenness 的图区域，根据 w1,w2,w3 上承载的 betweenness 流量和 v-$w_i$ 边的 betweenness 可以递归地计算出 sv 上的最短路径。

## 3.2. 聚类效果的度量和自动聚类分析

切片算法有个与生俱来的缺陷，就是需要事先指定一个切去的边的条数，或者是最终团伙的个数。同时，对于聚类分析的团伙结果，我们也需要做一个质量检测。因此，我们还需要一个检测团伙聚类质量的度量。M. E. J. Newman 和 M. Girvan 在其经典论文[1]中就提出一个聚类效果的度量：Modularity Qualify。这个度量通过计算团内联系和团外联系的比值得出一个衡量团伙聚集度的值。这

个值在 0~1 之间，值越大说明团伙的聚集度越高，即聚类的效果越好。通常，在实验中，Modularity 的值一般在 0.3~0.7 之间，非常高的值在实际数据中是比较罕见的。Modularity 度量也是本研究中的一个关键技术。

有了度量 ,我们可以对切边算法做进一步优化 ,使其自动化起来。通常来说，逐一切片，网络的 MQ 度量会呈现一个或多个峰值，最高峰值是理论上的最优聚类结果。我们对网络逐一切边，记录过程中的切边过程和 MQ 度量。以实现两个方式发现最优结果：a.贪心法找到第一个峰值结束；b.逐一切去所有边，找到最高峰。另外切边过程中记录过程值，可供使用者手工调整达到最优。

## 3.3. 时间片之间的动态事件发现

时间维度上的团体演化，可以分为以下几个基本动作，如 Figure 6 所示:1. 团伙成长；2. 团伙衰退；3. 团伙合并；4. 团伙分裂；5. 团伙出现；6. 团伙瓦解。对于个人来说，四个基本动作：1.出现;2.消失;1.进入团伙；2.离开团伙[3]

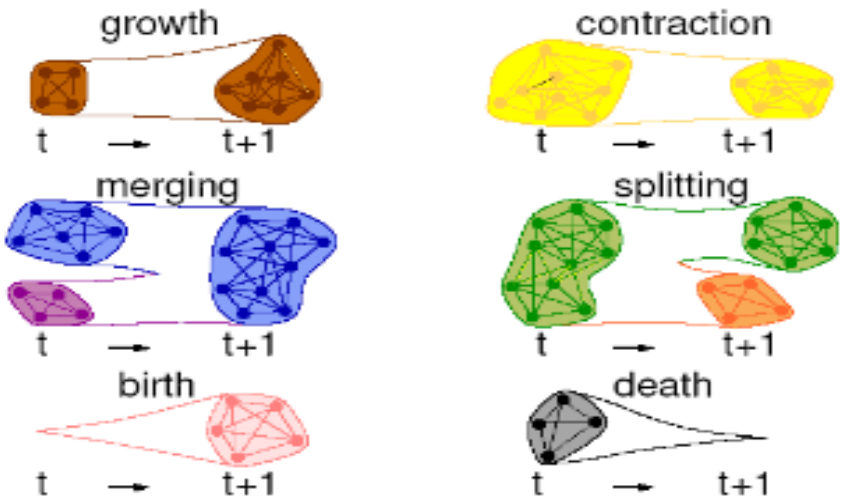


Figure 6：团伙在时间线上的基本事件

无论是个人行为发现还是团伙演化研究，其中一个基本点也是难点在于切片之间团体的对应关系的发现和保持[7]。简单解释就是在切片 t 上的团体 A0，到

了切片 t+1 上，我们要在聚类出的团体中找到和 A0 对应的团体 A1。如果发生的是分裂和合并还需要追踪到最为相近的一系列团伙。追踪的过程比较复杂也是动态挖掘的难点所在。研究的思路还是要化纷繁为简单，抓住核心。首先最基础的一个基线是找到最相近的团体。

要找到相近团体，最基本的是相似度的度量，我们使用 Jaccard 系数[6]进行度量。度量公式为：C1,C2 的相似度 = |C1 ∪ C2| / |C1 ∪ C2|

切片 t 中团体集合 St0={P0,P1,P2,…,Pn}，切片 t+1 中团体集合 St1={Q0,Q1,Q2,…,Qm} ,两两之间(Pi 与 Qj 间)都会有一个相似度 Jaccard 系数。对于单团，取出最大的相似度对应团是非常自然的想法。但是不能这样简单地处理，必须全局地看待这个问题。即取出全局上最优的匹配组合。这样，这个问题可以转化为二部图的最佳匹配问题。对于二部图的最佳匹配问题 ,我将采用 Kuhn - Munkras 算法[10]来解决二部图匹配问题。

对于个人行为，出现和消失都是比较显而易见的现象，直接查询存储的数据即知。而在时间轴上完成了对团伙的追踪后，只需要检测个人所属的团伙 id 也可方便地知道个人的进入团伙和离开团伙行为。

研究团伙的活动，对于团伙的出现和团伙的瓦解两个动作也是比较简单的了。直接查阅切片间的团伙 id 就可以知道。对于完全的团伙保持( 无任何成员进出 )，完全的团伙合并（子团完全合并，无成员溢出和加入），完全的团伙分裂（母团完全分裂成子团，无成员溢出和加入），也是比较简单的。比较相似团体的成员合并关系就可以直接得出结论。

但是，在实际情况中，团伙的行为并不总是完全的行为。还伴随着比较复杂的个体行为。团伙的演化行为虽然是比复杂的，而复杂的活动可能有几个基本活

动组合而成，其活动性质介于几种活动之间。这时候，要界定这种活动属于那种活动 ,最行之有效的方式是引入一个度量活动程度的值 k。这就有了如 k-Merge、k-Split 等一系列度量方法 ,来确定该活动应该属于哪种活动。其算法在 Sitaram Asur, Srinivasan Parthasarathy 等人的[3]一文中被提出来。我们将采用这种处理方式来分析团伙的动态行为。

# 4. 预期研究结果

1) 研究给出一套有效的对动态社会网络进行挖掘和分析的方法和算法
2) 开发实现出相关的算法，计划使用 java 为核心语言平台辅助以 xml、python、Shell 程序及 mysql 数据库实现相关的算法和数据结构
3) 基于 Java swing 技术开发出一套具有可视化界面的程序 ,把算法程序整合到可视化程序中 ,便于形象生动的表现动态网络的情况和算法的效果。
4) 使用多类型的实际数据集对算法进行验证实验。在实验中分析团伙的演化现象和规律，得出一些有意义的结论

# 5. 进度计划

目前开题报告的工作基本接近尾声。接下来的两个月的时间里，主要的工作就是对算法的实现和改进，以及使用实际的数据集进行验证和研究

**3.15-4.1:**

对整个软件进行设计，实现图的描述和基本的设计数据结构主要是动态图的描述。实现 betweenness 边算法，实现以 modules 聚类质量度量为依据的自

动切边聚类。实现静态部分的可视化软件开发

**4.1-4.15:**

实现二部图的匹配算法，建立模型分析时间切片间的图演化。并使用数据集验证，改进分析的算法。实现动态图分析的软件可视化部分

**4.15-4.30：**

使用数据集进行验证和研究在相关数据集中团伙的演化行为。继续优化算法和软件可视化部分。可扩展研究其他的如 core 动态分析方法的应用

**5.1-5.20：**

完成毕业论文，并且给出根据实验结果而得到的解释，以及一些算法复杂度分析

# 6. 参考文献


[1]. M.E.J. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (2) (2004) 026113.

[2]. M.E.J. Newman, From the cover: Modularity and community structure in networks, PNAS 103 (2006) 8577-8582

[3]. S. Asur, S. Parthasarathy, D. Ucar, An event-based framework for characterizing the evolutionary behavior of interaction graphs. In Proc. of KDD'07, pages 913-921, 2007

[4]. G. Palla, A.-L. Barabási, T. Vicsek, Quantifying social group evolution, Nature 446 (2007) 664-667

[5]. Nina Mishra, Robert Schreiber, Isabelle Stanton, and Robert E. Tarjan Clustering Social Networks, Computer Science, 4863 (2007), 56-67

[6]. Chayant Tantipathananandh, Tanya Berger-Wolf, David Kempe, A Framework For Community Identification in Dynamic Social Networks. In Proc. of KDD'07, pages 717-726, 2007

[7]. Tanya Y. Berger-Wolf, Jared Saia, A Framework for Analysis of Dynamic Social Networkss. In Proc. of KDD'06, pages523 – 528, 2006

[8]. U. Brandes, A faster algorithm for betweenness centrality, J. Math. Sociol. 25 (2001) 163-177

[9]. Santo Fortunato. Community detection in graphs. Physics Reports 486 (2010) 75-174

[10]. H. W. Kuhn. The Hungarian method for the assignment problem. Naval Research Logistics Quarterly 2 (1955) 83–97

**本科毕业论文（设计）文献综述**

# 动态社会网络团体发现与分析算法研究

# An Approach to Discover and Analyze Communities in Dynamic Social Networks

## 1. 文献综述

对于动态社会网络团体发现和团伙演化事件分析的算法研究，当前几乎所有的算法都采用先将动态网络按照时间线进行时间切片，再研究各时间片之间的演化过程。就像对影片的剪辑研究，需要先研究影片的各个帧。

将动态网络作时间切片之后，下一步的工作就是需要对各个切片上的团伙进行聚类。当前主流的研究在这点上可以分为两类：一是把每个切片上的网络看作静态的网络，进行聚类，挖掘出其中的团伙[3,4,11]；二是每个切片上对团伙聚类计算都需要考虑到前一切片，甚至前几个切片的状态。动态聚类的研究者认为，使用动态聚类的方法能够得出更为准确的聚类结果，另外利用前面切片的团伙情况可以减少后面切片的计算消耗。使用静态聚类的研究者认为，使用动态聚类并不能取得好的聚类效果。前面切片的聚类结果会影响后面切片的聚类结果，这使得每个切片的结果都引入了“脏”数据。特别是在动态网络发生重大事件的时候，如果受到前一稳定切片的不良影响，这个事件可能会被忽略，或者很难检测到。使用静态聚类的方式独立地分析各个切片是准确的。

## A. Hopcroft 的动态算法

最初的研究来自 Hopcroft 等人[11]，他们研究了来自 NEC CiteSeer 数据库的论文引文数据，这个数据集包含 1990 年~2001 年十余年之间的数据。他们使用层次聚类的方式来发现各个时间切片之间的团。在每个时间切片中，HopCroft 等人定义了一些“自然团”。所谓自然团，就是在聚类树中，只轻微地受到图的小扰动影响的团。小扰动指在图中移去很小一部分节点。自然团的定义类似于相关研究中提出的稳定团。Hopcroft 等人在研究使用最优匹配的方法来追踪不同时间片之间的团。使用这种匹配方式，也可以追踪某个团伙的整个历史演化过程。最优匹配方法也是我的研究中打算采用的方法。Hopcroft 的研究中，最主要的缺点就来自使用了层次聚类作为聚类算法。使用层次聚类将产生一个庞大的层次聚类树，这样使得图有非常多的划分方式，这不利于找出最优的划分。

Figure 7 展示了各个学科(领域)之间的研究交叉关系网络。这张网络就是根据科学研究的论文引文记录建模构建的。图中节点代表学科(领域)，边代表学科(领域)之间的交叉关联，边越粗代表关联越紧密。

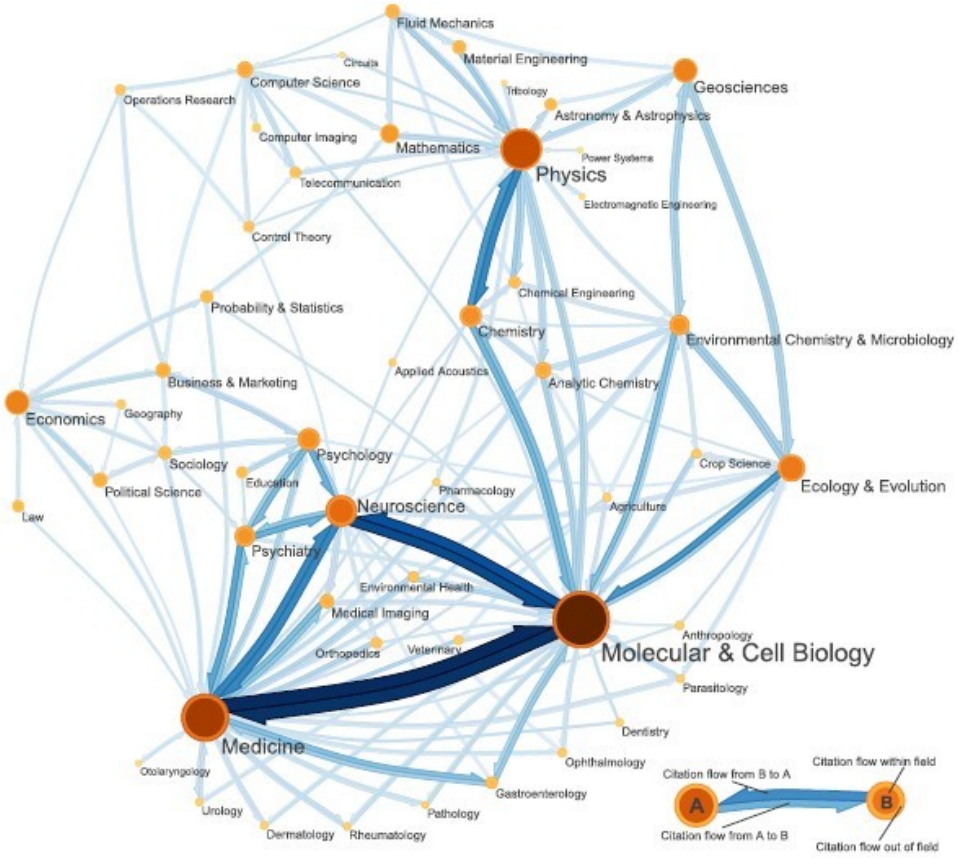


Figure 7：学科引文关系网络，表示不同学科之间的交叉引用关系。节点表示学科(或研究领域)，边代表在研究中的引用关系，边的粗细代表关联的紧密程度，越粗的关联度越大

## B. 基于事件(Event)的挖掘算法

Palla 等人也研究了一个比较系统的分析动态网络的算法[4]。他们使用以下两个数据集作为研究对象：(1)某电话公司一年中的用户电话通信记录(2)来自 Cornell 大学图书馆的包含 142 个月的文章作品合作者数据。Palla 等人的研究中采用了一种叫做 CPM 的方法(Clique Percolation Method)来进行聚类。CPM 聚类方式同时考虑两个时间切片的情况进行聚类。研究另一个中心是时间片之间的团伙匹配。一个最直接的想法就是将两个时间片交叉重叠，重叠部分最多的一对团伙，就是两个时间片上最接近的团伙。但是这样做是有问题的，例如在时间片 t 上，团伙 A1={1,2,3,4,5,6,7,8,9,10}，B1={11,12,13}，在时间片 t+1 上，A2={1,2,3,4,5,6,7 }，B2={8,9,10,11,12}，C2={13}。如果按照重叠最大匹

配来匹配团伙，那么 B2 在时间片 t1 上重叠最大的是 A1，这样子 B2 会被看做 A1 的延续，这样是不对的。所以他们将两个切片合并(merge)到一起，提出关联重叠度的定义。不单纯地考察重叠的面积。这个算法其实在早期的集合学中也被用作考量两个集合的相似度：Jaccard 系数。如公式(1)所示

$$\mathrm{J(t)} = \frac{|C(t_0) \cap C(t_0+t)|}{|C(t_0) \cup C(t_0+t)|} \qquad \text{公式(1)}$$

可以看出，计算这个度量的值非常快速，基本不需要消耗计算资源。准确、快速使得这个度量成为后来几乎所有研究中公认的一个度量两个切片间团伙相似度的算法。我的研究中也将使用这个度量衡量两个时间切片上的团伙相似度。

这个研究的另一个里程碑是提出预测网络在下一时间的状态的算法。Palla 等人利用前几个时间片的演化趋势信息来预测下一时间片可能发生的演化。如 Figure 8 所示，Palla 以一个节点对团伙内和团伙外的联系权值比来预测节点离开团体的可能性 p1。随着节点对外通信比例的增加，这个节点离开团伙的可能性增加，相对的，稳定性就减小。

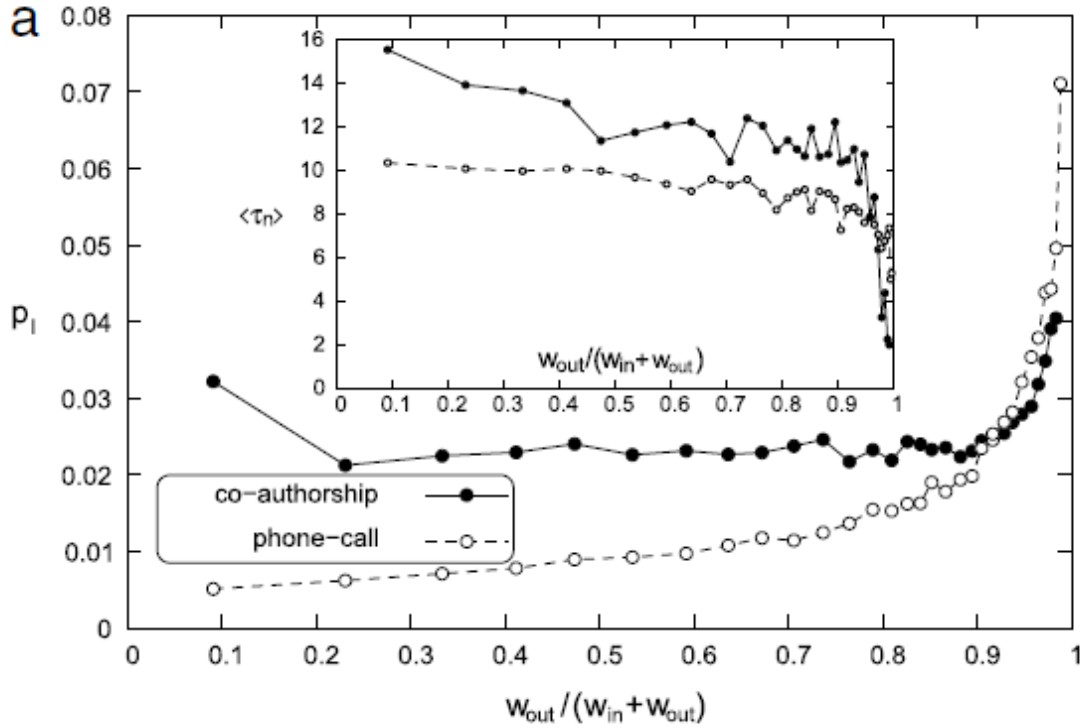


Figure 8：预测下一时间，某节点离开团伙的可能性指数

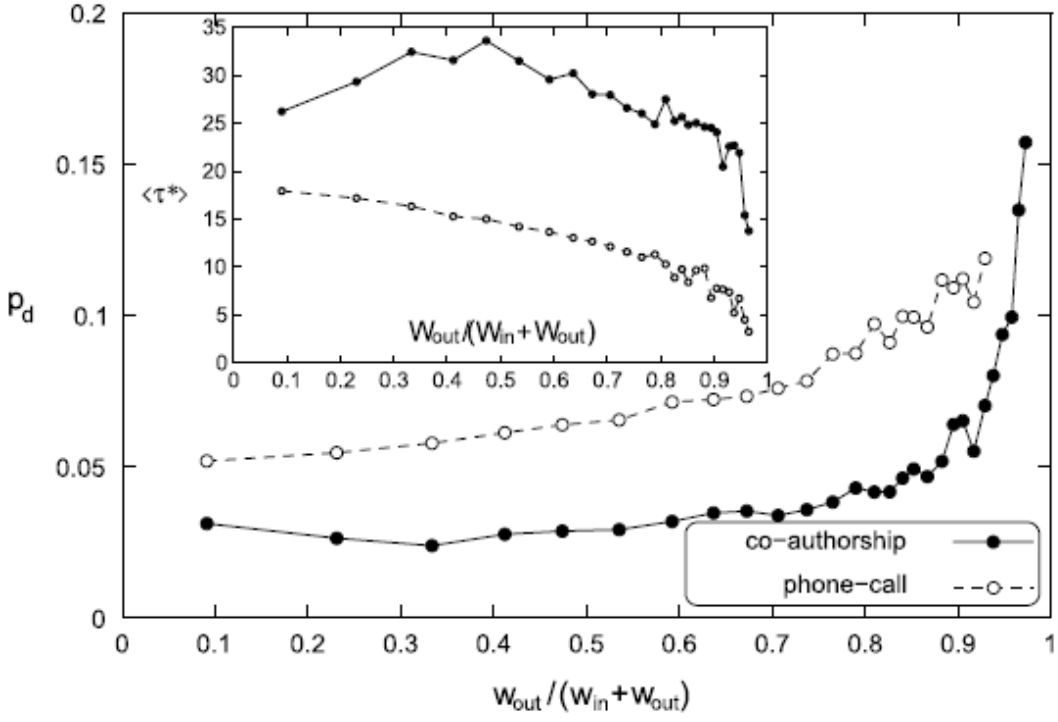


Figure 9：预测下一时间团伙发生分裂的可能性

同样的，Palla 也有研究一个团伙是否会发生分裂（或瓦解）的可能性。如 Figure 9 所示，随着一个团伙对外通信比例的上升，这个团伙发生分裂（或瓦解）

的可能性上升，相对的，这个团伙的稳定性就下降。

Asur 等人的研究[3]提出团伙的演化事件定量的定义，包括延续 Continue，合并 k-Merge，分裂 k-Split，生成 Form，分解 Dissolve。

[1]. 延续 Continue：团伙保持其全部的成员，当边的结构可以有所改变

[2]. 合并 k-Merge：时间片 t+1 中的某团伙$V_{t+1}^{j}$保有前一时间片 t 上的两个团伙的 k%的成员。显然，当 k=100 的时候，两个团完全合并成新团，没有成员逸散。

$$\frac{|(V_t^h \cup V_t^l) \cap V_{t+1}^j|}{Max(|V_t^h \cup V_t^l|, |V_{t+1}^j|)} > k\%$$

Formula 1: k-Merge 发生的条件公式

[3]. 分裂 k-Split :时间片 t 中的某团伙$V_t^j$在后一时间片 t+1 上分裂成两个团伙，两个团伙的成员总和要占原始团体的 k%以上。显然，当 k=100 的时候，这个团体完全分裂成两个团体，没有成员逸散。

$$\frac{|(V_{t+1}^h \cup V_{t+1}^l) \cap V_t^j|}{Max(|V_{t+1}^h \cup V_{t+1}^l|, |V_t^j|)} > k\%$$

Formula 2：k-Split 发生的条件公式

[4]. 生成 Form：在某时间片 t+1 中出现了团体 V，而团体 V 中的成员在前一时间片 t 上没有任何两个属于同一个团伙

[5]. 分解 Dissolve：在某一时间片 t 上的某一团伙 V，在下一时间片 t+1 分解，没有任何 2 个成员还属于同一时间片

Asur 还提出节点在动态图上的活动：

[1]. 出现 Appear：在时间片 t 上出现了前面时间片中没有的新节点

[2]. 消失 Disappear：在时间片 t+1 上，曾经在之前的时间片 t 上有的某节点消失

[3]. 加入 Join：以前不在团伙 V 中的某节点在这一时间片进入到这个团伙里

[4]. 离开 Leave：以前在团伙 V 中的某节点在这一时间片离开这个团伙

## C. 关键算法 Betweenness 和 Modularity

对于动态网络挖掘的一个核心问题：聚类算法的选择，我们有必要比较一下各种聚类的思路。聚类算法从过程上看，大致可以分为两类：1)起初是把网络看做一个团，然后依据某些规则将这个起初的大团进行划分，如 betweenness 切边算法；2)起初把网络中的每个节点看做一个团，然后依据某些规则进行聚合，如层次聚类。两个思路各有优劣，但根据本次研究的应用背景起源来看，选择划分法是显然较优的。在挖掘犯罪团伙的社会关系的应用场景中，往往是先已知犯罪团伙中的一个或多个成员。对全网络的聚类计算复杂度太大。首先会根据六度空间理论以这些已知成员做一次 k 度的扩展(k<6, 通常取 3 即可)，对扩展好的网络再进行分析，去掉“无嫌疑”的节点。使用划分算法可以只着眼于包含嫌疑成员的团进行分析，从而大量减少计算复杂度。另外独立的划分算法容易开发成为并行化的计算程序，由于划分聚类在不同团中的计算是相互独立的，所以可以方便将分析放置在多个进程(多处理器或多机)上进行并行计算。

### Betweenness 算法

M. E. J. Newman and M. Girvan 在其经典论文[1]中提出经典的 betweenness 算法。所谓 betweenness，就是最短路径 betweenness，这是边上的一个权重度量( 当然后续的研究者也在节点上设置 betweenness 度量 )。

分析网络中所有点两两之间的最短路径，统计通过某边 $E_i$ 的最短路径条数 b，就称作这条边上的 betweenness。而最短路径通量——betweenness 最高的边就是网络中的关键路径，这些边是连接团伙间的桥梁。将这些关键路径”切”去，就可以发现网络中的团伙。

Betweenness 聚类算法的流程如下：

1) 计算图中各边的 betweenness
2) 找出 betweenness 最高的边(如果包含多条相等最大的边就随机选一条)，并在团伙中去掉这条边
3) 重新计算网络的 betweenness
4) 从步骤 2)循环计算

每次循环切去一条边，切去指定的边数，或者团伙情况达到预期结果就停止计算。

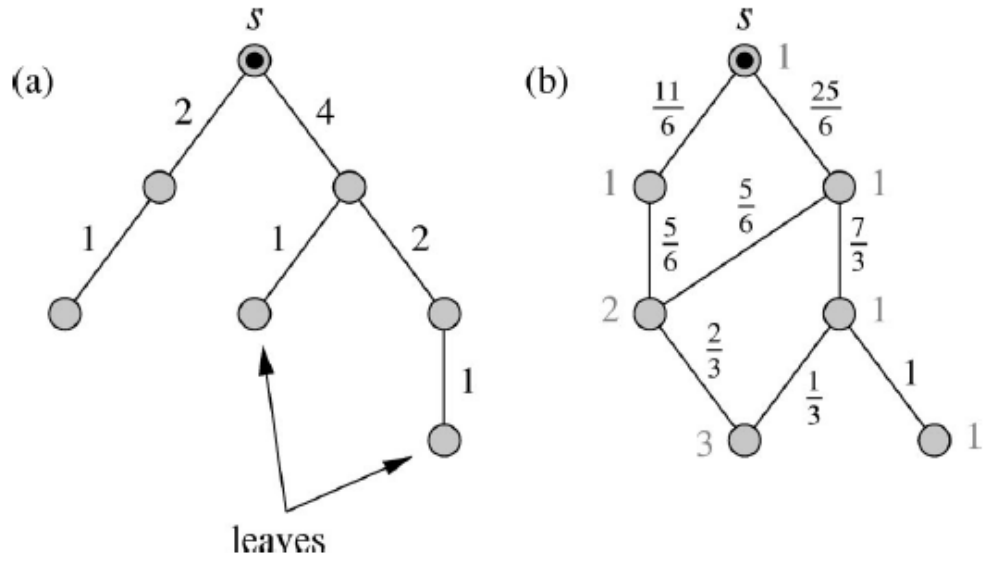


Figure 10：betwenness 计算的例子，从 s 节点出发到各点的 betweenness

可以看出，本算法的关键在于 betweenness 的定义。下面，我们以树形的网络结构为例叙述 betweenness 的定义，如 Figure 10(a)所示。边上的数字就是

从 s 点出发到各点的最短路径条数。对于一些复杂的图，两点间的最短路径可能不止一条，如 Figure 10(b)所示。节点边的数字代表从 s 点出发到该点的最短路径条数，边上的数字代表该边上的 betweenness。对于最短路径有 k 条的两节点，每条最短路径分流得到 1/k 的 betweenness，叠加计算到通过的边上。

假设一个图中有 m 条边，n 个节点。那么从一个节点出发做一次宽度优先遍历发现最短路径的时间复杂性是 O(m)。有 n 个节点就有 $O(n^2)$条最短路径。所以计算一次 betwenness 的时间复杂度是 $O(mn^2)$，这也是切去一条边的时间复杂度。

## Modularity 度量算法

M. E. J. Newman and M. Girvan 的论文[1]中还提出了 Modularity 的定义。关于 Modularity 的详尽定义在他们的另一篇论文[2]中被提出。一个网络被分为 k 个团伙，那么我们设置一个 k×k 的矩阵 **e**，其中的元素 $e_{ij}$ 表示，团伙 i 到团伙 j 的连接边占网络总边数的比例，再设 $a_i$ 是矩阵上第 i 行的元素和（也可求列和）。记 Tr **e** 为矩阵 **e** 的迹，即主对角线的和。记||**e**||为矩阵的元素和。那么 modularity 度量的计算公式的一种表达如下：

$$Q = \sum_i (e_{ii} - a_i^2) = Tr\ \boldsymbol{e} - ||\boldsymbol{e^2}||$$

Formula 3：Modularity 计算公式 1

Modularity 最初的定义是这样的，该定义也在[2]中提到。在一个有 m 条边，n 个顶点的图中，矩阵 **A** 是图的邻接矩阵，设 $k_i$ 是节点 i 的度数，$c_i$ 是节点 i 所在的团伙。$\delta(c_i,c_j)$是 kronecker delta 即克罗内克符号函数，那么 modularity 的计算公式如下：

$$Q = \frac{1}{2m}\sum_{ij}(A_{ij} - \frac{k_i k_j}{2m})\delta(c_i, c_j)$$

Formula 4: Modularity 计算公式 2

Modularity 的计算有多种推论公式，其中在 Ulrik Brandes 等人的论文[26]中就提出多种推到形式的 Modularity 计算公式。其中一种比较简洁的形式如下，其中 m 为网络中边的总条数，k 为团伙个数，$m_i$ 为团伙 i 内部边的条数，$d_i$ 为团伙 i 内各点的度数和：

$$Q = \frac{1}{m}\sum_{i=1}^{k} m_i - \frac{1}{4m^2}\sum_{i=1}^{k} d_i^2$$

Formula 5: Modularity 计算公式 3

利用 modularity 可以衡量聚类的效果。如 Figure 11 所示，对于同一个网络，不同的聚类结果具有不同的 Modularity 值。在(a)中，modularity 几乎是 0，在(b)中 modularity 的值接近于 0.5。

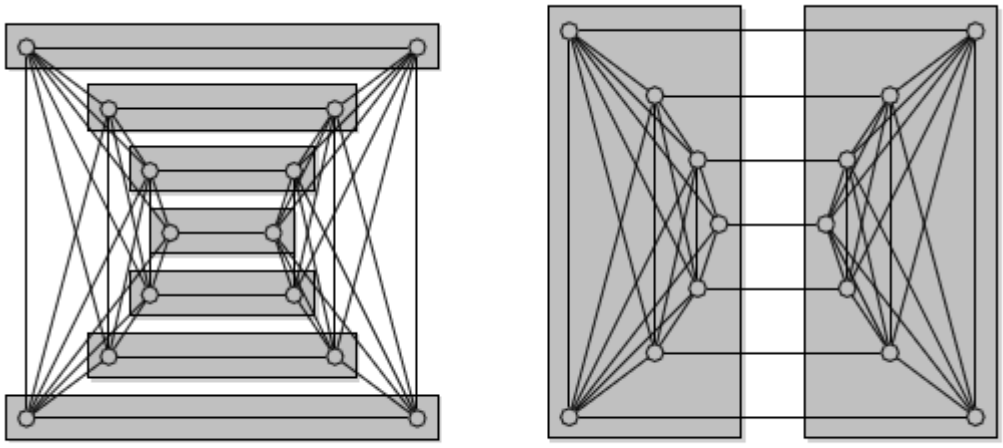


Figure 11: 对于同一个网络的不同聚类

Modularity 的值在 0~1.0 之间，在实际的网络图中，比较好的聚类结果，modularity 的值通常在 0.3~0.7 左右，较高的 modularity 值是很罕见的。

# 2. 参考文献


[1]. M.E.J. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (2) (2004) 026113.

[2]. M.E.J. Newman, From the cover: Modularity and community structure in networks, PNAS 103 (2006) 8577-8582

[3]. S. Asur, S. Parthasarathy, D. Ucar, An event-based framework for characterizing the evolutionary behavior of interaction graphs. In Proc. of KDD'07, pages 913-921, 2007

[4]. G. Palla, A.-L. Barabási, T. Vicsek, Quantifying social group evolution, Nature 446 (2007) 664-667

[5]. Nina Mishra, Robert Schreiber, Isabelle Stanton, and Robert E. Tarjan Clustering Social Networks, Computer Science, 4863 (2007), 56-67

[6]. Chayant Tantipathananandh, Tanya Berger-Wolf, David Kempe, A Framework For Community Identification in Dynamic Social Networks. In Proc. of KDD'07, pages 717-726, 2007

[7]. Tanya Y. Berger-Wolf, Jared Saia, A Framework for Analysis of Dynamic Social Networkss. In Proc. of KDD'06, pages523 – 528, 2006

[8]. U. Brandes, A faster algorithm for betweenness centrality, J. Math. Sociol. 25 (2001) 163-177

[9]. Santo Fortunato. Community detection in graphs. Physics Reports 486 (2010) 75-174

[10]. H. W. Kuhn. The Hungarian method for the assignment problem.

Naval Research Logistics Quarterly 2 (1955) 83–97

[11]. J. Hopcroft, O. Khan, B. Kulis, B. Selman, Tracking evolving communities in large linked networks, PNAS 101 (2004) 5249-5253.

**本科毕业论文（设计）外文翻译**

# An Event-based Framework for Characterizing the Evolutionary Behavior of Interaction Graphs

# 一种基于事件的挖掘动态图演化行为的算法框架

Sitaram Asur, Srinivasan Parthasarathy, and Duygu Ucar
Department of Computer Science and Engineering
Ohio State University
fsrinig@cse.ohio-state.edu

（注解，Interaction Graph的本意是交流图，在文中的定义是基于时序发生变化的图，翻译为动态图更易于理解。Characterize的原意是刻画描写，此处按照数据挖掘的概念惯例可以将其翻译为“挖掘”更易于理解）

## 摘要

动态图在很多领域是非常常见的，例如：生物信息学、社会学、物理学等等。尽管已经有很多关于挖掘网络图的研究，但这些研究大多都基于对静态图的挖掘研究上。我们这个研究将基于动态图在时间线的进化行为，研究图中的个体、团伙以及它们之间的信息流基于时序的变化历程。在这个研究中，我们采用事件刻画的方式来研究随时序变化的动态图的临界状态模型。我们的研究采用这个动态图上的时间切片，切片上的团伙之间不重叠。我们还研究出一种算法框架来发现和获取动态图的事件。我们使用这些基本事件来描述这个动态图中个体和团伙间复杂的行为模式。我们还证明了关于行为模式在模型进化、链接预测、影响最大化模型中的应用。最后，我们将基于我们的算法框架展示一个进化网络的扩散模

型。

## 1. 介绍

很多社会网络和生物学系统都可以抽象成一个复杂动态图。个体就是这个动态图中的节点 ,而个体间的联系就是图的边。对这些图的研究横跨多个科学领域，如计算机科学、物理学、生物学、社会学等。另外 ,互联网上的社区 ,例如 Flickr、MySpace、Orkut、Email 网络、共同作者网络、万维网等都是动态图的很好的例子。我们的这个研究可以提供一个研究它们的结构、性质和行为上的理解模型。

在这个领域上早期的一些研究局限于研究静态的网络，而忽视了真实世界的网络图大多是动态的。事实上，很多这样的动态图都是基于时序变化的，时序变化也包含边和节点的随时间的增加和减少。最近，开始出现一些关于动态图的研究。这些研究有这些主要组成部分：定义动态图的变化行为，描述这些行为，根据前期的变化预测后续的行为，对通用的图进化模型的研究。而这些研究通常是比较有挑战性的。例如，要研究一个快速增长的网络社区，就需要分析海量的时序数据来描述社区的结构、动态活动和演化历程。

自然中的动态图一般来说是可组合的。存在于点与点之间的联系可以用于将点集聚类为团伙。例如，在社会网络中，团伙代表着一群具有某种相似的关联或共同爱好的人。在过去的十数年中，这个问题在静态图上被很多人研究过了。但是，在实际的动态图中，一个固有的特性就是这些团伙不是静态的。它们随着时序时刻都在变化着或者说进化着。我们相信，研究图中的团伙的形成、演化、分解，对研究动态图的演化非常有意义。

另一个方面，网络中节点的行为也是至关重要的。在一个出在演化中的动态图中，节点的行为反应了它所代表的实体的联系模式随时序的变化。节点的运动、行为以及它们对其他节点的影响等信息可以帮助我们预测团伙的演化行为。例如，在一个社会网络中，假如一个人在社交上非常的活跃，那么他/她和其他成员建立新的联系的可能性就很大，也就具有很大的可能性去加入新的团体。在一个描述研究合作者的网络中，如果一个人经常和不同的人合作研究，那么当建立起一个新的合作项目时，包含这个人的可能性就比较高。某个节点的影响力可以具体化为研究它对其他节点的影响。如果在某一个个体加入一个团体时，有很多其他个体也同时加入，那么这个个体就具有比较高的影响力。

对于一个在演化着的动态网络中的传播流或者信息流的研究，对社会学研究、营销网络应用以及传染病学等都有非常重要的意义。例如，流行病毒对社会造成重大的威胁，因为它们很可能会广泛地传播，造成严重而广泛的疾病和伤亡。在营销中，营销的目的是通过联系网推广一种产品或者思想或者技术。研究动态网络的进化以及挖掘出网络中具有强影响力的个体（或团队），在实际应用中有非常重大的意义。例如帮助政府制定有效的政策，帮助传染病学者和防疫部门建立起有效的防疫隔离机制，帮助营销和广告业制定有效的营销手段。

在这篇论文里，我们将介绍一种挖掘演化中的动态网络的框架。我们首先将动态网络按照时序切片成为静态网络。然后我们独立地对各个切片中的网络进行聚类。紧接着，我们采用临界事件描述这些团伙的演化行为。我们将指出一系列高效的算法包括位图矩阵的计算等。我们采用关键事件来研究并发现动态图中行为导向的度量的剧变，这将是一个对于描述动态图中的演化行为的非常新颖有趣的方法。我们使用两个具体的示例来演示我们的挖掘框架——DBLP 共同作者网

络，一个患者临床实验网络。在每个实例中，我们采用我们提出的算法框架进行挖掘得出的行为模式都帮助我们建立了有用的关于团伙进化以及演化语言的推论。例如在 DBLP 数据集中，我们使用的未来趋势预测模式就取得了非常成功的结果。在临床实验网络中，我们将展示，使用我们的行为模式可以帮助我们分析发现某一种特定药物引起的肝毒性影响。最后，我们使用行为度量去详细分析一个动态图中的扩散模型，并且证明了这些模型在影响最大化作业中的应用。

简单地说，我们的研究重点如下：

- 发现动态网络中的关键事件
- 用于发现关键事件的有效的增量算法
- 一个基于我们的算法框架的扩散模型
- 事件和行为度量在两个动态网络真实数据集挖掘上的应用，预测行为、趋势及影响最大化。

## 2. 相关工作

关于动态图的兴趣模式挖掘（兴趣模式指社区发现和社区演化），学界早已表现出极大的兴趣。不过，这些相关的研究大多数是对静态图进行研究，挖掘出其中的团伙结构、模式以及突变信息。最近，关于动态图上的团伙的演化行为受到了几个研究团体的关注。Leskovec等人在进化的动态图上的研究，是基于图的拓扑结构的，例如度数(degree)的分布，团伙在整体网络上的性质等等。他们指出一个图的祖辈模型，称作forest fire（森林大火）模型，来描述他们在图的进化行为上的发现。Backstrom等人研究图的结构以及图的进化路径。为了判断

个体加入团伙的可能性，他们应用了决策树技术来研究个体和团伙的性质。他们也使用决策树来研究团伙增长的可能性。

Chakrabarti等人对于动态图的研究基于两种应用广泛的聚类算法——K-means聚类和层次聚类法。他们使用有效的聚类方式进行团伙划分，并与前一时间戳的团伙情况进行比较。在获取一个时间片上的聚类的时候，他们也考虑了历史聚类信息，即前面一些切片上的聚类情况。Falkowski等人使用团伙的子团伙的情况来研究一个团伙是活跃的还是稳定的。尽管他们也是研究动态图的，但是他们的研究思路和我们的有很大的区别。他们检测重叠的动态图时间切片，然后使用统计上的度量来描述子团的延续性。我们所关注的重点在于发现动态图中可识别的重要事件和行为模式，建模以及预测未来的发展趋势。在这点上，我们还特别关注了图中的节点并分析他们的进化行为。

Samtaney等人所作的一篇有很大影响力的论文中，描述了在二维或三维的矢量或标量域中提取相关局部用于追踪发展趋势。为了研究这些域在时间线上的进化历程，他们展示了对象的确定性进化事件。基于事件的方法也被应用在坐标数据和聚类流数据上。虽然他们使用了事件分析的方法，不过他们研究的不是动态图。

本体上的语义相似性的应用研究在过去已经被研究过多次。它已经被应用在很多的分类法上，例如Word-Net Gene Ontology。Resnuk提出一种新方法评估基于信息实体概念的语义相似性。在我们的研究中，语义相似性的概念被用于度量个体(数据集中是作者)之间和团伙之间的相似度。

## 3. 问题定义

在详细地介绍我们的基于事件的算法框架之前，请让我们先介绍一下本文中常用的一些标记。如我们之前提到的那样，我们所关注的主要是动态图的演化过程。特别是要研究动态图中的团伙和个体在时间线上的行为模式。为了更好地理解基于时间进化的图，很有必要先研究和描述一下图在时间线上的转换经历。在这点上，我们将使用动态图在不同时间上的时间切片来描述图的静态版本。

定义：当一个动态图的（内部）相互作用关系随时间发生了变化，我们就说这个动态图发生了演化。设G = （V，E）表示一个随时间变化动态图，其中，V为这个图中的点集（实体集），E为这个图的边集（实体间相互作用的集合）。设$G_i$ = （$V_i$，$E_i$）为G的一个时间切片，设[$T_{si}$，$T_{ei}$]是这个时间切片的时间间隔。时间切片的点集和关系集合是图G在这个时间间隔上聚集。

随着图的演化，新的节点和边可能加入到图中。同样的，节点和边也有可能消失。这样的一个在时间维度上具有动态行为的图可以描述为一系列时间切片的集合，其中，每个时间切片具有S equal（间隔等量），不相交（时间片中的团伙不重叠）的性质。

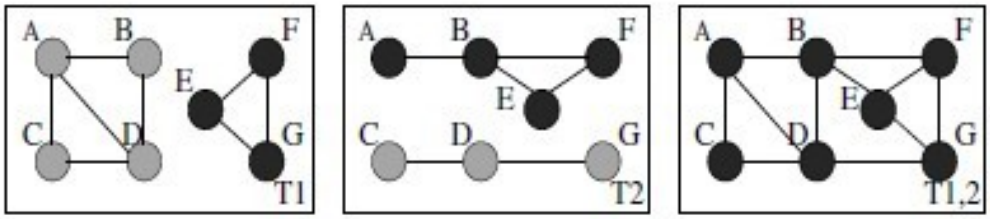


Figure 12：时间切片，从左到右分别是：a)时间切片 t=1；b)时间切片 t=2；c)时间累积切片 t=2

请注意，不同的时间切片是互斥的。它们不包含任何公有信息。这与之前的

一些研究不同，那些研究考虑前一个或一些时间片对本时间片的影响。Figure 12Figure 12：时间切片，从左到右分别是：a)时间切片 t=1；b)时间切片 t=2；c)时间累积切片 t=2 使用时间片展示了一个动态图的进化过程。我们可以发现，在时间切片 T1 中存在与 AC 之间 AD 之间的联系在后一时间片中不存在了。Figure 12 还展示了两个时间切片的累积值。我们发现在这个图中，刚才发现 AC 边和 AD 边消失的信息丢失了，比起前两个切片的表达也没有反映出图的真实情况。为了防止这种信息丢失，我们尽量选取比较小的时间间隔来做时间切片，然后选取比较有代表性的时间切片来研究。时间 T 下的所有时间切片的集合为 $S = \{S_1, S_2, \ldots, S_T\}$

要研究图的演化过程，我们还需要表示动态图在不同时间切片间的结构。为了达到这样的效果，我们需要先挖掘出每个时间切片中的团伙。设在时间切片$S_i$中共有$k_i$个团伙，这些团伙的集合为$C_i = \{C_i^1, C_i^2, \ldots, C_i^{ki}\}$。对于时间切片$S_i$中的第j个团伙，$C_i^j$也可以看做一个图，（$V_i^j$, $E_i^j$），其中$V_i^j$是团伙中的节点集合，$E_i^j$是团伙中成员的联系集合（边的集合）。于是，对于每一个$S_i$ = （$V_i$, $E_i$）来说，$V_i^1 \cup V_i^2 \cup \ldots \cup V_i^{ki} = V_i$

为了选择一个有效的聚类算法，我们实验了大量的聚类算法，并使用很多的动态图来检测这些聚类算法在团伙聚类效果上的表现。（modularity是度量聚类效果的一种度量值，这里我都翻译成聚类质量）。我们发现MCL算法角力非常好的聚类效果，而且它在不同时间片之间也非常稳定，并且它具有很好的自动性。因此，我们觉得使用MCL算法来对不同的时间片进行聚类。MCL算法不需要指定需要聚类成的团伙数量。相对的，它使用一个颗粒度参数来确定划分的数量。因此，对于每个时间切片来说，这个切片上的聚类效果很大程度上依赖于节点在这个切片上的联系情况。对于我们用于实验的比较稀疏的图来说，我们采用颗粒度1.2来进行聚类。

Algorithm 1展示了我们的算法框架的基本思路。我们遵循增量的策略来挖掘团伙信息，找到动态图随时间演化过程中比较明显的变化(时间片中比较明显的变化)，并和关键事件建立起联系。这些关键事件在后续的研究中将被用于分析更加复杂的行为模式。在本文的第五部分，我们会叙述这些关键事件，以及我们怎样发现这些关键事件。在本文的第六部分，我们挖掘出这些关键事件，并用于分析复杂的行为模式。

Algorithm 1：Mine-Event(G, T)　　算法1：事件发现算法

Input(输入): 动态图G = (V, E)，时间T，时间间隔

首先将动态图G划归为时间切片的表示：

$S = \{S_1, S_2, \ldots, S_T\}$

**for** i = 1 to T **do**

    Cluster $S_i$

    $C_i = \{C_i^1, C_i^2, \ldots, C_i^{ki}\}$

**end for**

**for** I = 1 to T – 1 **do**

    Events = find_events($S_i$, $S_{i+1}$)　　//Section 5

    Mine Events for complex patterns　　//Section 6

**end for**

Algorithm 1

## 4. 数据集

在我们的研究中，我们采用了以下两个数据集：

## DBLP 合作作者网络

DBLP目录库（数据库）中保存了超过80万个计算机科学家的发表的论文。我们选取DBLP的数据中的一些重要会议和研究领域的论文数据来生成了一张合作作者的网络图。我们选取的研究领域主要是数据库技术、数据挖掘技术、人工只能技术领域。我们选取了10年间（1997~2006）横跨这三个领域的28个研究方向的所有论文。我们把这些论文生成了一张网络图，其中，每个作者就是这个图中的一个节点，而作者之间的合作关系就成为这个图中的边。于是，这个横跨十年的图包含了23136个节点，以及54989条边。我们选择时间切片的大小是一年，从而产生了10个时间切片。这些切片就被用作研究关键事件和模式。我们确信，研究DBLP合作作者网络的进化将为我们研究提供有利的信息，帮住我们研究自然状态下的合作关系演化，以及预测作者之间未来可能的合作。

## 临床实验数据

在临床实验中，制药商需要研究某种药物的效果以及药物的副作用，以确保这种药物是否能在可忍受的副作用范围能产生有效的医疗效果。

（以下省略部分段落）

## 5. 临界事件

(省略部分段落)

团伙事件:我们定义了五种基本事件，基本事件定义于连续时间片上的团伙。

设$S_i$和$S_{i+1}$为时间片集合S上的两个连续时间片，它们中包含的团伙集合分别为$C_i$和$C_{i+1}$。以下是这五种基本事件的定义：

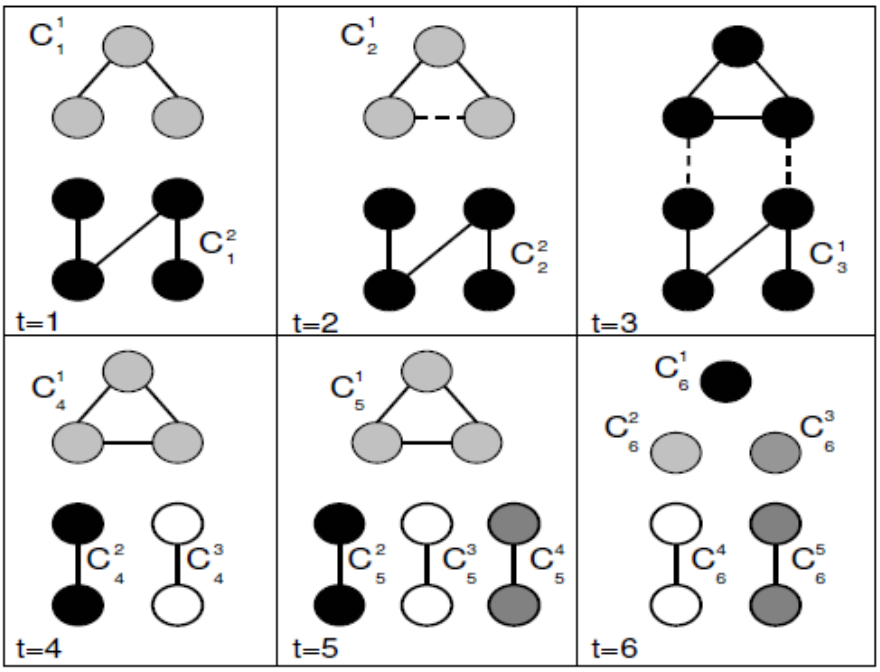


Figure 13：一个网络图在 6 个时间片 t=1 到 6 上的时间片。不同的团伙使用不同的颜色标出。

## 1) 延续 Continue

当团伙$C_{i+1}^j$内的点集$V_{i+1}^j$与$C_i^k$上的点集$V_i^k$相同时，我们说这个团伙发生延续事件。我们不需要关心边的变化：

$$\text{Continue}(C_i^k, C_{i+1}^j) = 1 \text{ iff } V_i^k = V_{i+1}^j$$

提出这个定义的动机是：如果一个团伙的成员一直保持，那么团伙中任一点的信息都可以最终到达另外任一点。也就是说，只要团伙点集保持不变，那么信息流不会收到影响。边的增减只对团内节点的关联度产生影响。如 Figure 13 所示 t=1 和 t=2 时间片上，两个团伙都发生延续事件。特别注意到，在$C_2^1$上虽然增加了一条边(联系)，但团伙并没有发生改变。

## 2) K-合并 k-Merge

两个不同的团伙$C_i^k$和$C_i^l$发生 k-合并事件，当且仅当在下一时间片上，存在一个团伙至少保有了这两个团伙里 k%的节点。必要条件就是:

Merge($C_i^k$, $C_i^l$, k) = 1 iff ∃$C_{i+1}^j$满足：

$\frac{|(V_i^k \cup V_i^l) \cap V_{i+1}^j|}{Max(|V_i^k \cup V_i^l|,|V_{i+1}^j|)} > k\%$ 并且 $|V_i^k \cap V_{i+1}^j| > \frac{|C_i^k|}{2}$ 并且 $|V_i^l \cap V_{i+1}^j| > \frac{|C_i^l|}{2}$

这种情况下，在时间片 i+1 上，$V_i^k$和$V_i^l$之间必然是存在着边(联系)的。直观地说，合并表达的就是在原先两个团伙之间形成了新的联系。这使得原先 2 个团伙中 k%的成员构成了新的团伙。可以看出，存在着一种完全的合并事件，也就是当 k=100 时，两个团伙的成员完全合并到下一时间片上的一个团伙中。在这种情况下，原有的两个团伙完全消失。如 Figure 13 所示，t=3 时就发生了完全合并。虚线指示的边新创建出来，所有的节点都合并到$C_3^1$中。

## 3) K-分裂 k-Split

某个团发生了分裂事件，当这个团中 k%的成员在下一时间片分裂到两个不同的团伙中。其必要条件为：

Split($C_i^j$, k) = 1 iff ∃$C_{i+1}^k$, $C_{i+1}^l$满足：

$\frac{|(V_{i+1}^k \cup V_{i+1}^l) \cap V_i^j|}{Max(|V_{i+1}^k \cup V_{i+1}^l|,|V_i^j|)} > k\%$ 并且 $|V_{i+1}^k \cap V_i^j| > \frac{|C_{i+1}^k|}{2}$ 并且 $|V_{i+1}^l \cap V_i^j| > \frac{|C_{i+1}^l|}{2}$

直观上来说，就是这一时间片上相关节点间的联系在下一时间片上被打断了，导致这些节点在下一时间片加入(组成)两个团。需要注意到，一个边的断开，并不一定意味着发生了分裂事件，因为还可能存在其他的联系保持了这个团伙的构成（和 k-连通性的概念类似）。在 Figure 13 中，t=4 时刻就发生一次完全的分裂事件，原团完全分裂为 3 个团伙。

## 4) 生成 Form

一个新的团伙$C_{i+1}^{k}$发生了生成事件，当且仅当这个团伙的任意两个节点在前面的时间片中不同属于任一团伙中。或者说，没有任何两个节点在$V_{i+1}^{k}$是在 t=i 时间片上的同一团伙的。

Form($C_{i+1}^{k}$) = 1 iff ∃! $C_i^j$有 $V_{i+1}^{k} \cap V_i^j > 1$

直观地说，生成代表着一个新团伙的建立。在 Figure 13 中，t=5 时刻就出现了 2 个以前没有的新节点，并发生了团伙的生成事件。

## 5) 分解 Dissolve

一个团伙发生了分解事件，当且仅当在下一时间片上，这个团伙中的任意两个节点都不同属于任一团伙。或者说，起始团伙中的任意两个节点间都不将存在联系。

Dissolve($C_i^k$) = 1 iff ∃! $C_{i+1}^j$有 $V_i^k \cap V_{i+1}^j > 1$

直观地说，在某一时间点上，一个团伙内部缺乏足够的联系，那么这个团伙就发生分解了。这意味这一个团伙(团队)的解体。Figure 13 中 t=6 上指示了一次分解事件，当团伙$C_5^1$中的三个节点不再有联系时，这个团伙分解成 3 个团伙