|  |  |  |
| --- | --- | --- |
|  | 0019M0et0(1000T00110TD100u1orcfo0 | 01610( |
|  | 105010 0(0E0(t0nerein Dr00.010C⁰ :  2D i05 010701040101010 | **01**00T0C 小 |
| D4ororupuie100010er 值 0 | 3401010.0t01卷1a001010rc 00101Qr | a00391010it |

**第** **8** **章** **社交网络分析**

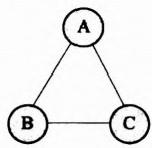
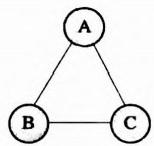
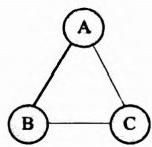
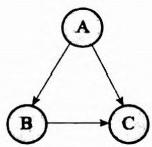
**8** **.** **1** **简** **介**

社交网络由节点和连接 (Ties) 构成。节点指的是网络里的行动者 (Actor), 连接则 表示这些行动者之间的关系。社交网络可以表达个人之间、团队之间、组织之间、计算机 之间、URL 网络地址之间以及其他各种实体之间的关系。这些实体也可以是一些概念。 社交网络可以绘制成一张图，图论是社交网络分析的基础。

在本章中，根据论述的需要，“节点”“顶点”“行动者”可以互换使用，“边”“连接” 和“链路”可以互换使用，“图”和“社交网络”也可以互换使用。

根据节点间的关系是单向的还是双向的，社交网络分为两类，分别是无向 (Undirect- ed) 社交网络和有向 (Directed) 社交网络。根据节点间的关系是有强度的还是没有强度 的，社交网络分为有权重 (Weighted) 的社交网络和没有权重 (Unweighted) 的社交网 络。这个权重在实际应用中代表不同的意义。比如在通信网络中，可以表示网络设备之间 的链路的传输速率。

需要注意的是，这两个分类角度不是互斥的，即它们是可以组合的。比如在无向的有 权重的社交网络中，节点之间只要有连接，就可以双向互相寻访得到，其连接是有权重 的。图8-1展示了社交网络的基本类型。



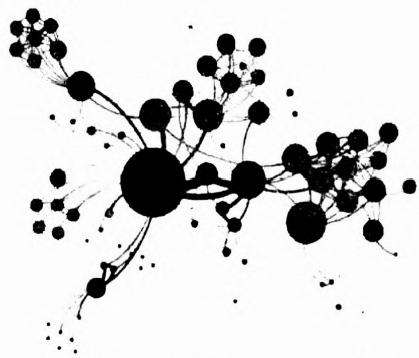
(a)Undirected (b)Directed (c)Weighted (d)Unweighted

**图** **8** **-** **1** **社交网络的类型**



图 8 - 1 (b) 是一个有向图，对于 A 节点来讲它和B,C 都有连接，连接的方向是 A→B.A→C, 对于C 来讲，它和A 和 B 都有连接，连接方向是A→C,B→C 。 一个节点 的连接数称为一个节点的度 (Degree), 根据方向进一步划分为出度 (Out Degree) 和入 度 (In Degree)。A节点的度为2,出度为2,入度为0,C 节点的度为2,出度为0,入度 为2。

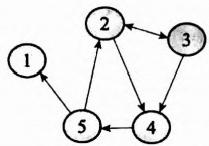
社交网络分析是对社交网络里的实体之间的关系、信息在实体之间的流动等方面，进 行度量和分析，使人们获得对整个网络的理解。社交网络分析的结果，以可视化的方式进 行展示，方便人们观察。比如把社交网络社区检测的结果显示出来时，不同社区的节点用 不同的颜色来表示.如图8-2所示。



**图8-2** **社交网络的社区检测和可视化(不同颜色/灰度代表不同杜区)**

资料来源：<https:/iwww.iit.demokritos.gr/group/persona.>

社交网络可以使用三种方式进行表示 (Representation), 分别是邻接矩阵 (Adjacen- cy Matrix)、边列表 (Edge List)、邻接关系列表 (Adjacency List)。图8 - 3为 一 个有向 图，它的三种表示方式分别列示如下。



**图8-3** **一个简单的有向图**

图8-3的邻接矩阵表示为：

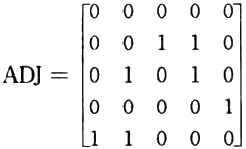




图8-3的边列表表示为：

|  |
| --- |
| 2,3  2,4  3,2  3,4  4,5  5,1  5,2 |

图8-3的邻接关系列表表示为：

|  |
| --- |
| 1:  2:34 3:24 4:5 5:12 |

**8.2** **社交网络分析的应用**

在现实世界中，大型社交网络无处不在，表达各式各样的关系，比如组织和国家之间 的贸易关系、论文引用关系以及共同作者 (Co-Authorship) 关系、电话呼叫关系、蛋白 质交互关系、交通网络所表达的航班关系或者街道关系、亲属关系、朋友关系 (Facebook 网 站 ，Twitter 网站)等。

社交网络分析的应用非常广泛，对上述关系构成的网络进行分析，使得我们获得对网 络的微观、中观、宏观等层次的理解。比如在经济活动中，利用社交网络分析，可以有效 地进行市场营销 (Marketing) 、 风险管理 (Risk Management) 和欺诈检测 (Fraud Detec- tion) 等。

具体来讲，在市场营销方面，我们可以预测客户的流失 (Customer Churn)、发动 市场营销活动 (Launching Marketing Campaign)。通过社交网络分析，我们可以对客 户进行划分，了解每个客户群组 (Group) 的行为特点，群组的特点有可能影响到客 户的流失率。我们可以基于这些分析结果，预测和避免客户流失。在发动市场营销活 动之前.如果能够了解到每个客户群组里具有较大影响力的客户 (Influencer), 就可 以针对这些客户展开营销，通过他们影响其他群组成员，以达成购买。在风险管理方 面，以股票为例，可以把股票组织成一个社交网络.用股票之间的价格变化的相似性 定义股票之间的连接，利用社区检测算法，把股票划分成多个群组，从每个群组里选 择股票，构造多样化 (Diversified) 的投资组合，从而避免一荣俱荣、 一损俱损的状 况发生。

8.3 社交网络分析方法



8.3.1 网 络 的 一 些 基 本 属 性

对于社交网络，我们首先关注的是整个网络的规模，即网络包含多少个节点 (Num- ber of Nodes) 以及多少条边 (Number of Edges) 等。除了这两个简单的指标之外，还需 要计算一些关键的指标，以对整个网络有所把握。这些指标具体解释如下。

1. 平均的度 (Average Degree)

表示整个网络中，每个节点平均有多少条边。换句话说，就是每个节点平均和多少个 节点有关系。其计算公式为：

Dag=(#of Edges)/(#of Nodes)

2. 半 径 (Diameter)

要了解社交网络的半径，需要先了解一个重要的指标，即测地距离 (Geodesic Dis- tance) 。 这个距离指的是两个节点间的最短路径 (Shortest Path) 上连接的数量，也就是 边的数量。

任意两个节点间，比如从节点i 到节点j, 可能有多个路径。路径的长度定义为从节 点i 到节点j 的连接的数量，也就是边的数量。在这些路径中，其中有一条或者若干路径 是最短的，称为最短路径或者测地距离 (Geodesic Distance)。比如在图8 - 4中，从节点 A 到节点D, 可以有ABD,ABCD,ACD,ACBD 等路径，其中ABD 和 ACD 的路径长度

都是2,是这些路径里最短的。寻找最短路径的算法将在“可达性和最短路径” 一节中 介绍。

网络的半径指的是在社交网络中任意两个节点间的测地距离的最大值。网络的半径给 我们一个直观的把握，到底网络有多大，也就是从网络的一侧到另外一侧，最多需要经过 多少个节点间的连接。比如图8-4中的社交网络，其半径为2。

B

A

图8-4 网络的半径

人们也可以使用网络的半径作为一个上界，在我们从某个节点开始探索 (Explora- tion) 网络时.限定需要走过的最大的节点间连接的数量。

3. 平均路径长度 (Average Path Length)

网络的平均路径长度指的是任意两个节点间的最短路径长度的平均值。它的计算公 式为：



式中，d, 为节点i 和节点j 之间的最短路径长度；N 为网络的节点数量。

4. 网络密度 (Network Density)

网络密度描述的是在所有可能的连接里，有多大部分是实际存在的。“可能的连接” (Potential Connection) 指的是网络上两个顶点之间可能存在的连接，不管它是否真的存 在。“真实的连接”(Actual Connection) 则是真实存在的连接。

为了对网络密度有个直观的理解，这里举两个例子。在家庭聚会中，对于所有可能的 连接，在两个聚会成员之间真的可能存在互相认识的关系，在这个网络中，网络密度是相 当高的。在一辆公共汽车上，两个乘客之间互相认识的可能性是非常低的，在这个网络 中，网络密度是比较低的。

从网络密度的定义可以看出，通过把实际存在的连接除以可能的连接数量，就可以计 算出网络密度。如果网络是无向图，可能的连接数量就是N×(N-1)/2 。 表 8 - 1 给 出 了几个简单的实例，计算网络(可以用无向图表示)的密度。

表8-1 计算网络密度实例

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | | C |
| B  A | A | B  C | B  A D |
| Nodes:2  Potential Connections:1 Actual Connections:1  Network Density:100% | Nodes:3  Potential Connections:3 Actual Connections:3  Network Density:100% | | Nodes:3  Potential Connections:3 Actual Connections:2  Network Density:66.67% |

5. 集聚系数 (Clustering Coefficient)

在图论中 (Graph Theory), 集聚系数用来刻画顶点聚集 (Cluster) 在一块儿的程度。 有很多证据显示，在现实世界的很多网络中，顶点具有形成紧密联系的群组 (Tightly Knit Group) 的倾向。集聚系数有两个版本，分别是全局 (Global) 集聚系数和局部 (Local) 集聚系数。

全局集聚系数基于三元组 (Triplet) 进行计算。三元组由互相连接的顶点构成。 一个 三角形包含三个封闭的三元组 (Closed Triplet),分别以三个顶点为核心。全局集聚系数 的计算公式为：

Clustering Coefficient=Number of Closed Triplet/Number of Triplet(Closed+Open)

比如，图8- 5中，ij-v 是一个封闭的三元组 (Closed Triplet.Triangle),m-v-n 是一 个开放的三元组 (Open Triplet)。把一个社交网络中所有开放的三元组和封闭的三元组都 找出来，利用上述公式.就可以计算集聚系数。

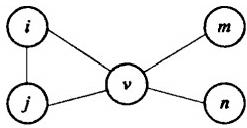


图8-5 封闭三元组、开放三元组与全局集聚系数



局部集聚系数描述的是某个顶点的邻居顶点，在多大程度上接近于形成一个完全图 (Complete Graph)。假设一个社交网络表示为一张图G=(V,E), 其 中V 为图的顶点集 合，E 为这些顶点之间的边的集合。边e; 连接vi 和v; 两个顶点。v;的近邻顶点集N; 是与v; 直接连接的邻居节点的集合，也就是 N={v,:e;∈E V en ∈E} 。k为 N, 的顶点个数， 即 k;=|N,I。

那么,有向图的顶点的局部聚集系数为 。无向图

的顶点的局部聚集系数为 

在此基础上，我们可以计算网络的平均集聚系数 (Network Average Clustering Coef- ficient.Watts 和Strogatz提出),这个指标可以作为全局集聚系数的一个替代指标，表示

整个网络的顶点的集聚程度。其计算公式为：,其中n 为网络中的顶点数量。

8.3.2 复杂网络的 一 些拓扑特性

1. 小世界网络模型 (Small World Model)

1967年，哈佛大学的社会心理学教授Stanley Milgram(1933—1984),试图描绘一个 连接人与社区的人际关系网络。他发现了六度分隔 (Six Degrees of Separation) 现象。 “你和任何一个陌生人之间，所间隔的人不会超过6个。也就是说，最多通过6个人，你 就能够认识任何一个陌生人”。这是一个非常有趣的现象。六度分隔理论说明，在社会中 普遍存在“弱纽带”,但是发挥非常强大的作用，这些“弱纽带”把人与人之间的距离变 得非常“相近”。

其他研究者对这个问题也进行了研究。其中，Watts 撰写了著名的社会学和社交网络 方面的著作 Six Degrees:The Science of a Connected Age 。1998年 ，Watts 和 Strogatz 在 Nature 杂志上发表文章，创建了 Watts-Strogatz 模型。这是一个具有 Small World 性 质的随机图 (Random Graph) 的生成模型 .这些网络具有较短的平均最短路径长度 (Short Average Path Length) 和较高的聚集系数 (High Clustering Coefficient)。

此外，Jon Kleinberg 对这个问题进行了形式化，转换成一个可以评估的数学模型，发 *表在其论文* *The Small-World Phenomenon:An Algorithmic Perspective 中。在日常生活* 中，我们认识的新朋友，可能有大家一起认识的人，于是感叹“世界真小”,Jon Klein- berg 的模型证实了这一点。

2. 幂律模型 (Power Law Model)

度指的是网络中顶点与其他顶点的关系(边)的数量。网络的度分布指的是从网络中 随机抽取一个顶点时，与这个顶点相连的顶点数(该顶点的度d) 的概率分布。换句话说， 就是某个网络中一个顶点拥有k 条边(有向边和无向边)的概率P(k)>0(k=0.1,2,…)。

比如，由 n 个顶点组成的完全图 (Complete Graph), 度的分布是d=n-1 的概率为1, 其余度的概率是0。

*1999年，Barabasi 和Albert 在Science 上发表文章* *Emergence of Scaling in Random* Netuorks. 指出许多实际的复杂网络的连接度的分布具有幂律形式。

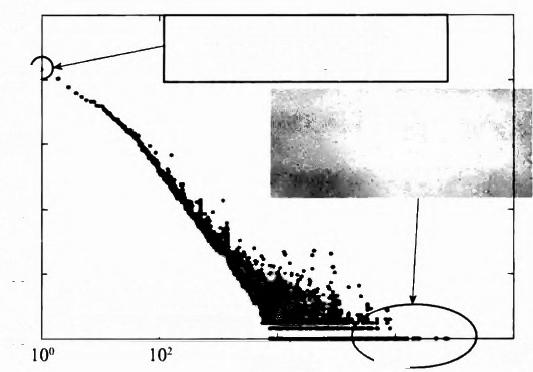
符合幂律 (Power Law) 模型的度分布指的是P(k) 具有这样的形式，P(k)=ak⁻。

幂律具有尺度不变性 (Scale Invariance), 也就是对自变量缩放一个常数c, 将引起函数值

*成比例缩放，P* *(ck)=a(ck)⁻=c⁻a k⁻=c'P(k)∞P(k) 。* *图8-6展示了符合幂律模*

型的一个网络数据集 (AltaVista WebGraph) 的度分布图。

Number of Vertices



10¹0

More than 10⁸vertices have one neighbor.

10

High-Degree Vertices

10⁴

10²

alta Vista WebGraph

1.4B Vertices,6.6B Edges

10⁶ 10⁸

Degree

10°

10⁶

**图** **8** **-** **6** **符合** **Power Law 模型的网络的度分布**

资料来源：<http://www.graphanalysis.org/SIAM-CSE13/08_Guestrin.pdf.>

无标度网络 (Scale Free Network) 是指度分布服从幂律的网络(或者至少是逼近 (Asymptotically) 幂律的)。上述公式中的γ值一般在(2,3)区间内。无标度网络中， 顶点之间的连接(度数)体现严重的不均匀分布特性，网络中少数顶点拥有极多的连接， 大多数顶点只有很少量的连接 。

Barabasi 和 Albert 给出了生成随机的无标度网络的Barabasi-Albert 模型 (Barabasi- Albert Model)。在 BA 模型中，当有新的顶点加入到网络中时，它根据概率和网络中的其 他顶点连接，新的顶点X 和老的顶点Y 相连的概率正比于Y 的度，也就是新的顶点更倾 向于连接到原本度数就很高的顶点，这个现象也称为“优先连接”(Preferential Attach- ment) 特性以及“富者更富”(Rich Get Richer) 或者“马太效应”(Matthew Effect)。通 过反复添加新节点 ， 就能构建出一个无标度网络 。

**3.** **其他网络模型**

除 了Barabasi-Albert 模 型 和Watts-Strogatz 模 型 ， 值 得 关 注 的 模 型 是Erdos-Renyi 模 型。①1960年，匈牙利数学家Erdos 和 Renyi 建立了随机图理论，提出了随机拓扑模型 Erdos-Renyi 模型。该模型一直是研究复杂网络的基本模型。它以两种方式进行描述：

(1)第一种描述方式：给定网络顶点总数N, 网络中任意两个顶点以概率P 连接，生成的 网络全体记为G(N,P), 构成一个概率空间。由于网络中连接数量是一个随机变量X, 取 值可以从0到 N(N-1)/2, 有 n 条连接的网络数目为CN(N-1)/2。可生成的不同网络的总数 为 2N(N-1)/2,它们服从二项分布。网络中的平均连接数也就是平均度为P\*N(N-1)/2。

(2)第二种描述方式：给定网络顶点数N 和连接数n, 这些连接是从总共N(N-1)/2 条 可能的连接中随机选取的，生成的网络全体记为G(N,n), 构成一个概率空间。可生成的

① 读者可以参考文献[http://doktori.bme.hu/bme\_palyazat/2012/hallgato/honlap/szabo \_david\_en.htm](http://doktori.bme.hu/bme_palyazat/2012/hallgato/honlap/szabo_david_en.htm). 获 得对Watts-Strogatz,Barabasi-Albert以及 Erdos-Renyi等网络模型的直观认识。



不同网络的总数为CN(N-1)/2, 它们出现的概率相同，服从均匀分布。

**4. 连通分支/连通分量** **(Connected Component)**

在一个大规模网络中，并非所有顶点都连通，可能存在多个连通分支。比如在社交网 络网站中，新注册的用户还没有加好友，那么他就是一个孤立的顶点(还没有和其他顶点 相连)。一旦这个用户有了好友，那么他就连接到大规模的社交网络中了，不再是一个孤 立的顶点。又比如，某个校园社交网络中，规定只能加本校的好友，不能加外校好友，那 么不同学校的学生就构成一个个独立的连通分支。

(1)强连通分支 (Strongly Connected Component)。在一个强连通分支中，每个顶点 和任意其他顶点通过一系列有向连接 (Directed Link) 到达，也就是任意两个顶点之间是 互相可达的，即对于任意顶点对A 和B, 既可以从A 到 达B, 也可以从B 到 达A。

(2)弱连通分支 (Weakly Connected Component)。在一个弱连通分支中，每个顶点 和任意其他顶点，可以从某个方向到达，也就是两个顶点，只需要有一个方向到达即可， 即对于任意顶点对A 和 B, 可 以 从A 到达B, 或者可以从B 到达A。

(3)大型分支 (Giant Component)。真实网络中，通常存在一个最大的连通分支，它 包含了网络中大多数的节点(比如超过80%的节点),这样的分支称为大型分支 (Giant Component) 。 在社交网络网站上的用户， 一般处在一个超大的连通分支中，真正孤立的 顶点或者小的连通分支所占的比例是很小的。比如，Facebook 的数据显示，约99.7%的 用户处在一个超大的Giant Component中。

**8.3.3 节点的中心性** **(Centrality)**

中心性用来量化一个节点在网络中的重要性。早在1978年， L.C.Freeman 就提出用 Degree,Closeness,Betweenness 等指标衡量社交网络节点的中心性。这些指标都是通过 节点在网络中的位置 (Network Location) 进行评估的。通过这些指标，我们可以了解到 在整个网络中，谁是连接者 (Connector) 或者领导者 (Leader), 谁是社区之间的桥梁 (Bridge) 等，甚至可以在社交网络中找到谁是专家 (Maven)。

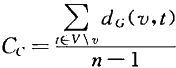
**1. 度中心性** **(Degree Centrality)**

度中心性的计算公式为 ,其中deg(v) 为节点v 的 度 ，n 为网络的节 点数量。从公式可以看出，度中心性基于顶点的度(也就是顶点的邻居个数)计算。度数 高的顶点，一般也称为Hub 顶点，Hub 顶点对于网络信息扩散有很大帮助，它更容易充 当信息交换的角色，比如无线通信中的基站、无线网络中的接入点等。

**2. 接近中心性** **(Closeness Centrality)**

接近中心性的基本思想是，如果一个顶点到其他顶点的距离都比较短，那么它处于网

络的中心。接近中心性可以用节点间距离 (Geodesic Distance, 也就是两个节点之间最短 路径中所包含边的数目)来表征。 一个节点到其他节点的最短路径都很短，那么该节点的 接近中心性就高。接近中心性指标可以用来衡量信息从某个节点传输到其他节点需要的时

间长短。接近中心性的计算公式为 ,dG(v,t) 表示节点v 和节点t 之间的 最短路径。



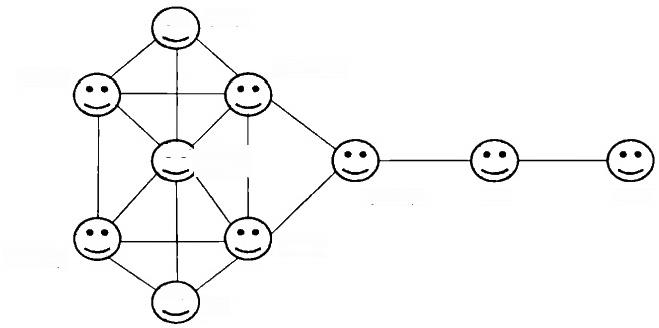
3. 中介中心性 (Betweenness Centrality)

如果网络中的某个节点，经常出现在其他节点间最短距离路径中，即其他任意两个节 点间最短距离的路径上经常包含该节点，那么该节点更有可能促进其他节点的通信和交流。

中介中心性的计算公式为 , 其 中 ，gk 为节点j 到节点k 的最短 路径，g;(v) 表示经过节点v 的节点j 到节点k 的最短路径。

我们不仅可以针对顶点计算中介中心性，而且可以针对各个边计算中介中心性。中介 中心性高的顶点/边，起到信息传输桥梁的作用，通常处于两个社区之间。

Kite Network 是由著名的社交网络研究专家David Krackhardt 构造的一个旨在说明重 要概念和指标的虚构网络。网络结构如图8-7所示。



Carol

Fernando

●

Diane

Heather

Garth

Ed

Ike Jane

Beverly

Andre

●●

●

图8- 7 Kite 网络 (Kite Network)

对这个网络的各个节点的中心性进行计算，如表8-2所示。

**表8-2** **网络各个节点的中心性**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Degree | | Closeness | | **Betweenness** | |
| Diane | 0.667 | Fernando | 0.643 | Heather | 0.389 |
| Fernando | 0.556 | Garth | 0.643 | Fernando | 0.231 |
| Garth | 0.556 | Diane | 0.600 | Garth | 0.231 |
| Andre | 0.444 | Heather | 0.600 | Ike | 0.222 |
| Beverly | 0.444 | Andre | 0.529 | Diane | 0.102 |
| Carol | 0.333 | Beverly | 0.529 | Andre | 0.023 |
| Ed | 0.333 | Carol | 0.500 | Beverly | 0.023 |
| Heather | 0.333 | Ed | 0.500 | Carol | 0.000 |
| Ike | 0.222 | Ike | 0.429 | Ed | 0.000 |
| Jane | 0.111 | Jane | 0.310 | Jane | 0.000 |

从上述中心性的计算结果我们可以看到，针对三个中心性的节点排名是不一样的。

首先来看度中心性。在Kite 网络中， Diane 和其他节点有最多的连接，使得她成为网 络里最活跃的节点，她是网络里的连接者 (Connector) 或者集线器 (Hub) 。 人们一般认 为，在社交网络里，和其他节点有更多的连接意味着节点越重要，其实这不尽然。在上述 网 络 中 ，Diane 只是和她的直接隶属的集群 (Cluster) 或者社团 (Clique) 里的其他节点 有连接，然而这些节点之间早就互相连接在一起了，没有 Diane, 他们也能够进行通信。



**对于中介中心性来讲，Heather的得分最高。** **Heather 处在网络中一个关键的位置上，** **它连接了整个网络的两个重要区域** **(Constituency), 扮演了一个中介** **(Broker) 的角色。**

对于接近中心性来讲，Fernando 和Garth 的得分最高，他们拥有到达其他所有节点的 最短路径。或者换句话说，他们和其他任何节点最靠近。于是这两个节点处在网络上一个 绝佳的位置，他们可以监控整个网络的信息流动 (Information Flow),他们更加容易看到 网络上正在发生什么事情，因为他们和其他节点的距离最短。

除了上述中心性之外，还有随机游走中心性 (Random Walk Centrality),由于实际应 用中主要用到的是上述中心性，在此不予赘述。

不同的中心性从不同角度度量顶点的重要性。这些中心性可以用于如下场合：(1)在 恐怖分子网络中，标定关键的行动者 (Key Actor);(2) 在生物网络中 (Biological Net- work) 考察蛋白质之间的相互作用 (Protein-Protein Interaction);(3) 在交通网络中 (Transportation Network), 标定关键的路段；(4)在社交网络中考察疾病的传播等。

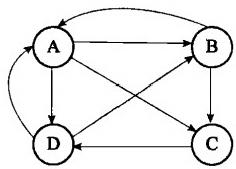
除了利用上述各种中心性 (Centrality) 来度量节点的重要性之外，还可以利用 Pag- eRank 算法和HITS 算法，评价节点的重要性。

**4.Page Rank 算法**

Google 公司早期的核心业务是搜索。 Google 能够击败其他对手成为业界的领导者， 得益于它的结果排序优于其他搜索引擎。 Google 根据结果的重要性进行排序，用户需要 的信息总是排在靠前的位置。在搜索结果排序中， Google 使用了上百个(100十)指标， 其中PageRank 扮演了重要的角色。

Web的页面数量巨大，用户提交一个检索请求得到的结果条目非常多，达到上万甚至 上百万条。好的搜索引擎必须想办法将最相关的、重要的结果排在前面。

PageRank 算法是衡量网页重要性的算法。互联网的网页之间通过超链接 (Hyper Link) 形成一张网络，网络中的每个节点代表一个网页 (Page), 这个网页可以通过链接 指向其他网页，当然也有从其他网页指向本网页的链接。为了说明 PageRank 算法，我们 使用四个网页A,B,C,D 构成的超链接图，如图8-8所示。



**图8-8** **A,B,C,D 四个网页及其之间的超链接**

对于某个网页来说，该网页的PageRank 计算基于两个基本假设：(1)数量假设。在 Web图模型中，如果一个页面节点接收到的其他网页指向的入链数量越多，那么这个页面 越重要。(2)质量假设。指向页面A 的入链质量不同，质量高的页面会通过链接向其他页 面传递更多的权重。因此，越是质量高的页面指向页面A, 则页面A 越重要。利用这两个 假设，PageRank 算法首先赋予所有网页相同的重要性得分，经过迭代计算，更新每个网 页的PageRank, 直到得分收敛为止。

在用户浏览网页时，他们通过超链接实现页面跳转，于是我们可以通过分析超链接组 成的拓扑图来推算每个网页被访问的频率的高低。最简单的做法是，当用户处在某个网页 时、他跳转到其他网页的概率是一样的。在图8-8中，A 页面有超链接指向B,C,D,

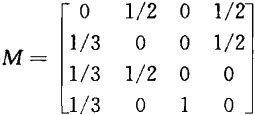


那么用户从A 页面跳转到B,C,D

的概率各为1/3。



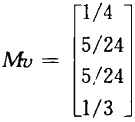
一般地，假设有N 个网页，我们可以建立一个 N\*N 的矩阵，矩阵的<i,j> 元素 的取值，表示用户从页面j 跳转到页面i 的概率，该矩阵称为转移矩阵 (Transition Ma- trix) 。 图8-8对应的转移矩阵为：



我们给每个网页一个初始的Rank, 分别是1/4,那么初始的Rank 向量为v=[1/4, 1/4,1/4,1/4],T 表示转置。

M 的第一行分别是A,B,C 和 D 转移到页面A 的概率.v 的第一列是A.B,C 和 D 当前的Rank 。因此用M 的第一行乘以v 的第一列，所得结果就是页面 A 最新 Rank 的 一 个新的合理估计。这里体现的思想是， 一个页面的重要性由所有链向它的页面(链入页 面)的重要性来决定，到一个页面的超链接相当于对该页投一票。

那么矩阵M 和向量v 相乘的结果，就分别代表A.B,C,D 新 Rank。



我们迭代这个过程 v=Mu, 即不断地用M 乘以新的 Rank 向量，又产生一个新的 Rank 向量。可以证明 Rank 向量 v 会收敛，这时迭代过程可以终止。最后， 一个页面的 PageRank由所有指向它的页面(链入页面)的重要性，经过上述迭代算法得到。

最终的v 向量就是各个页面的PageRank 的值。比如 ·上面的向量经过几步迭代之后， 收敛到<1/4,1/4,1/5,1/4>,这就是A,B,C,D 页面最后的PageRank。

计算网页的 PageRank 算法，有一系列问题需要处理。

5. 解决 Dead End 问题

网络中有一类节点，它们不存在外链，称为Dead End 节点。比如图8-9中节点D 就 是 Dead End节点。

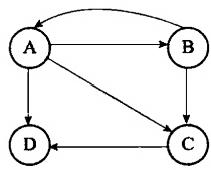


图8-9 Dead End节点

在这个图中，M 的第四列将全为0。在没有Dead End的情况下，每次迭代后向量v 各 项的和始终保持为1.有了Dead End, 迭代结果将最终归零。显然 ·这是不合理的。

解决办法是不断拿掉图中的Dead End 节点及Dead End 节点相关的边，直到图中没有 Dead End。对剩下部分计算各个节点的PageRank, 然后以拿掉Dead End的逆向顺序，反

推各个 Dead End的 PageRank。

比如针对图8-9,我们拿掉D, 再拿掉C, 剩下 A 和 B, 此时可很容易算出A,B 的 PageRank 均为1/2。按照逆序反推C 和 D的 PageRank, 由 于C 前置节点有 A 和B,A 和 B的出度 (Out Degree) 分别为3和2,因此C 的 PageRank 为：1/2×1/3+1/2×1/2= 5/12。同理，D 的 PageRank 为：1/2×1/3+5/12×1/1=7/12。

6. 解决 Spider Trap问题

假设网络中存在这样的页面，它有外链，但是这个外链指向自己，这种节点称为 Spi- der Trap。如果使用上述迭代算法计算 PageRank. 会导致 Spider Trap 节点的PageRank 越来越大，趋近于1,其他节点的PageRank 的值几乎为0。显然，这是不合理的。

为了解决这个问题，需要对PageRank 计算方法进行平滑处理，具体做法是加入“远 程转移”(Teleporting) 。 远程转移就是在任何一个页面浏览的用户，都有可能以一个极小 的概率，直接转移到另外一个随机页面.无须这两个页面间存在超链接。换句话说，就是 在任意时刻，用户到达某个页面后，不再从该页面的链接向后浏览的概率。迭代公式改

为：。式中，β往往被设置为一个比较小的参数(0.2或更小);e 为 N 维单位向量。按这个公式迭代算下去，Spider Trap的效应被抑制了，从而保证每个页面 都获得一个合理的PageRank。

7. 话题敏感 (Topic-Sensitive) 的 PageRank

不同用户可能对不同的话题感兴趣。话题敏感的PageRank 预先定义几个话题类别， 为每个话题单独维护一个向量，然后想办法了解用户的话题倾向，根据用户的话题倾向， 对结果进行排序。我们可以使用Open Directory(DMOZ)(<http://www.dmoz.org/)> 的一级话题类别作为话题分类 (Topic)。

目前 DMOZ 的 一 级 Topic 有 ：Arts (艺术)、 Business (商务)、Computers (计算 机)、Games (游戏)、 Health (医疗健康)、 Home (居家)、 Kids and Teens (儿童)、 News (新闻)、Recreation (娱乐休闲)、 Reference (参考)、 Regional (地域)、 Science (科技)、Shopping (购物)、Society (人文社会)、Sports (体育)等。

建立了话题分类以后，需要将网页归入最合适的分类。分类方法包括自动方法，比如 基于单词的TF-IDF 分类或者人工方法。然后分话题，计算网页的PageRank 。Topic-Sen-

sitive PageRank的迭代公式为  。单位向量e 变为了s,s 是这样一 个向量，如果某个网页在 Topics中，则s 中第k 个元素为1,否则为0。对于每一个Topic 都有一个不同的s, |s| 表示 s 中1的数量。

当用户提交搜索查询时，确定用户的话题倾向，选择合适的 PageRank 向量，以对网 页进行排序。确定用户话题倾向的方法包括让用户主动选择其主题倾向，或者通过跟踪用 户的搜索行为.建立用户画像，刻画用户话题倾向等。

8.PageRank 的计算性能及其应用

互联网的网页数量巨大，假设有10亿个网页，那么上述迭代算法中的M 矩阵有100 亿亿(十亿乘以十亿)个元素，矩阵相乘的计算量非常大。 Larry Page和 Sergey Brin两 个人利用稀疏矩阵计算的技巧，大大简化了计算量。

值得指出的是，PageRank 不仅用在网页重要性的排序中，也可以用在其他网络中计 算节点的重要性。

**9.HITS 算法**

HITS(Hyperlink Induced Topic Search) 算法是康奈尔大学 (Cornell University) 的Jon Kleinberg 博士于1997年提出的。这个算法是网页链接分析中非常重要的算法之一。

Authority 页面(权威页面)和Hub 页面(枢纽页面)是HITS 算法最基本的两个定 义。Authority 页面是指与某个领域或者某个话题相关的高质量网页。比如，在搜索引擎 领域，Google 、百度的首页就是该领域的高质量网页。在视频领域，YouTube 、优酷、土 豆等视频网站的首页就是该领域的高质量网页。Hub 页面指的是包含了很多指向高质量 Authority 页面链接的网页。

HITS 算法的目的是通过一定的迭代计算过程，在海量的网页中找到和用户查询相关 的高质量Authority 页面和Hub 页面。其中，Authority 页面能够满足用户查询的高质量 内容。搜索引擎以这些网页为搜索结果，返回给用户。HITS 算法的基本思想是， Au- thority 页面和Hub 页面具有相互增强的关系。(1)一个好的Authority 页面会被很多好 的 Hub 页面指向；(2)一个好的Hub 页面会指向很多好的Authority 页面。

算法的执行过程包括三大步骤，分别是构造根集合、扩展根集合以及迭代计算各个网 页的Authority 和 Hub 得分。

(1)构造根集合Root 。将查询 Q 提交给基于关键字查询的检索系统，从返回结果页 面集合中取前n (比如n=200) 个网页，作为根集合 (Root Set), 记为Root 。由此可见， Root 中的网页数量较少，这些网页是和查询 Q 相关的网页，Root 中包含较多的权威 (Authority) 网页。这些网页以及网页之间的链接关系构成了一个有向图G(V,E)。

(2)扩展根集合 Base。在根集合Root 的基础上，对网页集合进行扩充，构造集合 Base。扩充过程描述如下：凡是与根集内网页有直接链接指向关系的网页都被扩充到集合 Base。也就是说，无论是有链接指向根集内页面，还是根集页面有链接指向的页面，都被 扩充进人扩展网页集合Base 。HITS 算法将在这个扩充网页集合内寻找好的 Hub 页面与 好的Authority 页面。

(3)计算扩展集合Base 中所有页面的Hub 得分(枢纽度)和Authority得分(权威度)。

|  |
| --- |
| ao=1,ho=1//a,h 为向量，各个元素初始化为1,分别对应一个网页的a/h 得分 t=1// 迭代时间步为1  do  For eachv in V {    at=a/|atl h=h/|h|  t=t+1  }  }while(|a-a1|+|h-h₁|>e) Return(a,h) |

图8-10给出了迭代过程中，某个页面的Hub 得分和 Authority 得分的更新方式。假 设以A(i) 代表网页i 的 Authority 权值，以H(i) 代表网页i 的 Hub 权值。比如.在扩充 网页集合中，有3个网页有链接指向页面1,同时页面1有3个链接指向其他页面，那么 网页1在此轮迭代中的Authority 权值，即为所有指向网页1页面的Hub 权值的总和。类

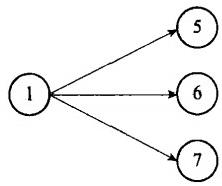
似地，网页1的Hub Hub

② ③

④

权值，即为所指向的页面的Authority 权值的总和。

Authority Hub Authority



①

A(1)=H(2)+H(3)+H(4) H(1)=A(5)+A(6)+A(7)

**图8-10** **Hub得分和** **Authority得分的更新方式**

HITS 算法不仅应用在搜索引擎领域，在社交网络分析等其他领域也得到了借鉴和使 用，取得了很好的应用效果。当然 HITS 算法也有一些不足，这需要我们在实际应用中加 以注意。这些不足包括：(1)结构不稳定。当在已有的扩充网页集合内添加或者删除个别 网页，或者改变少数链接关系，那么HITS 算法的排名结果就会发生较大的改变。(2)计 算效率不高。首先HITS 算法必须在接收到用户查询后实时进行计算，此外，该算法需要 进行多轮迭代计算才能获得最终结果，计算效率不高。(3)主题漂移问题。如果在扩展网 页集合里包含部分与查询主题无关的页面，而且这些页面之间有较多的相互链接指向， HITS 算法很可能会给予这些无关网页很高的排名，导致搜索结果发生主题漂移，这种现 象称为紧密链接社区现象 (Tightly Knit Community Effect) 。(4) 容易被作弊者操纵排名 结果。如作弊者可以建立一个网页，页面内容增加很多指向高质量网页或者著名网站的网 址，这就是一个很好的Hub 页面，之后作弊者再将这个网页链接指向作弊网页，于是可 以提升作弊网页的Authority得分。

**8.3.4** **可达性、路径、最短路径、最小生成树**

**1.** **可达性** **(Reachability)**

网络里顶点之间的可达性指是否存在从一个顶点到另外一个顶点的路径。一个顶点A 是 从另外一个顶点B 可达的，是指从B出发经过一系列的连接能够到达A 而不管中间要经过多 少连接。从顶点B 到顶点A 的一系列连接，称为从节点B 到节点A 的一条路径 (Path) 。 同理 .从顶点A 到顶点B的一系列连接，称为从节点A 到节点B的一条路径 (Path)。

如果网络对应一个有向图，那么有可能存在这样的状况，即我们可以从顶点A 到达顶 点 B, 却不能从顶点B到达顶点A, 具体的例子包括网页中的超链接 (Hyperlink)。

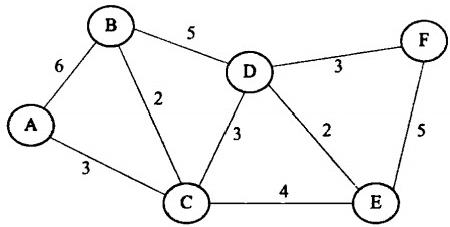
在实际网络中，从一个顶点到另外一个顶点， 一般存在多条路径 (a Number of Paths) 。 在从顶点A 到顶点B 的所有路径中，我们关注最短路径。

**2.** **最短路径** **(Shortest** **Path)**

最短路径的概念已经在“网络的一些基本属性”一节中进行介绍，这里介绍具体的寻

找最短路径的算法。寻找最短路径最经典的算法是Dijkstra (迪杰斯特拉)算法。该算法 是典型的单源最短路径算法，用于计算一个节点到其他所有节点的最短路径。它以起始点 为中心，向外层层扩展，直至扩展到终点为止。

该算法的具体思路为，设G=(V,E) 是一个带权有向图，V 为顶点集合，E 为边的集 合。首先，把图中顶点集合V 分成两组，第一组为已求出最短路径的顶点集合，用S 表 示。初始时S 中只有一个源点，以后每求得一条最短路径。就将它加入到集合 S 中，直到 全部顶点都加入到S 中，算法就结束了。第二组为其余未确定最短路径的顶点集合，用U 表示，按最短路径长度的递增次序依次把第二组的顶点加入S 中。在加入的过程中，有一 个约束条件，总保持从源点v 到S 中各顶点的最短路径长度不大于从源点v 到 U 中任何顶 点的最短路径长度。此外，每个顶点对应一个距离。S 中的顶点的距离就是从v 到此顶点 的最短路径长度。U 中的顶点的距离，是从v 到此顶点只包括S 中的顶点为中间顶点的当 前最短路径长度。图8-11为一个无向图.图中每个边上标注的数字表示边的长度。



**图8-11** **Dijkstra 算法实例**

现在要求出从A 到其他各个顶点的最短路径，Dijkstra 算法的各个步骤如表8-3所示。

**表8-3** **Dijkstra算法的步骤**

|  |  |  |
| --- | --- | --- |
| 步骤 | S集合 | U集合 |
| 1 | 选择A, S=<A>  最短路径A→A=0  以A为中间点，从节点A开始找 | U=<B,C.D,E,F> A→B=6  A→C=3  A→其他U中的顶点=∞ 发现A→C=3为最短 |
| 2 | 选择C,这时S=<A,C>  最短路径A→A=0,A→C=3  以C为中间点，从A→C=3这条最短路径开 始找 | U=<B.D.E.F>  A→C→B=5(比第一步的A→B=6短，这时 候，把B的权值改为A→C→B=5)  A→C→D=6  A→C→E=7  A→C→其他U中的顶点=○  发现A→C→B=5为最短 |
| 3 | 选择B,这时S=<A, C .B>  最短路径A→A=0,A→C=3,A→C→B=5  以B为中间点，从A→C→B这条最短路径开 始找 | U=<D.E.F>  A→C→B→D=10(比第二步A→C→D=6要  长，这时到D的权值，更改为A→C→D=6) A→C→B→其他U中的顶点=∞  发现A→C→D=6为最短 |



续前表

|  |  |  |
| --- | --- | --- |
| 步骤 | S集合 | U集合 |
| 4 | 选择D . 这时S=<A . C,B .D>  最短路径A→A=0.A→C=3,A→C→B= 5,A→C→D=6  以D为中间点，从A→C→D这条路径开始找 | U=<E,F>  A→C→D→E=8(比第二步A→C→E=7要  长，这时到E的权值，更改为A→C→E=7) A→C→D→F=9  发现A→C→E=7为最短 |
| 5 | 选择E,这时S=<A, C,B, D, E>  最短路径A→A=0,A→C=3,A→C→B= 5.A→C→D=6,A→C→E=7  以E为中间点，从A→C→E=7这条最短路 径开始找 | U=<F>  A→C→E→F=12(比以上第四步的A→C→ D→F=9要长，这时F的权值更改为A→C→ D→F=9)  发现A→C→D→F=9权值最短 |
| 6 | 选择F,这时S=<A, C,B,D, E, F> 最短路径为A→A=0,A→C=3,A→C→  B=5,A→C→D=6,A→C→E=7,A→C→ D→F=9 | U集合已空，查找完毕 |

寻找最短路径，除了Dijkstra 算法，还有Floyd-Warshall 算法。它是一个经典的动态 规划算法，能够寻找任意两点间的最短路径，可以处理有向图或带负权的图的最短路径问 题，也用于计算有向图的传递闭包。该算法的时间复杂度为()(N³), 空间复杂度为 O(N²) 。 具体的算法流程，请参考相关资料。

需要注意的是，从节点i 到节点j 的最短路径可能有多条。我们一般假设所有的信息/ 影响力 (Information/Influence) 仅仅沿着最短路径传播 (Flow/Propagation) 。 但是，在 实际应用中，信息的传播有可能沿着某个最短路径，也可能沿着接近最短路径的其他路径 (Near-Shortest Path)。即便节点i 和j 是相邻的，从节点i 到节点j 的信息传播有可能是 直达的 (Direct), 也可能是通过其他节点到达目标节点j 的 (Indirect)。

**3. 最小生成树** **(Minimum Spanning Tree)**

给定一个无向图，如果它的某个子图中任意两个顶点都能互相连通并且是一棵树，那 么这棵树就称为生成树 (Spanning Tree)。如果边上有权值，那么使得这些边权值之和最 小的生成树称为最小生成树 (Minimum Spanning Tree,MST) 。MST 在交通网、电力 网、电话网、管道网等设计中均有广泛的应用。最小生成树可以使用 Prim 算法或者 Kruskal 算法计算。下面仅介绍 Prim 算法，Kruskal 算法的详细信息，请读者参考相关 资料。

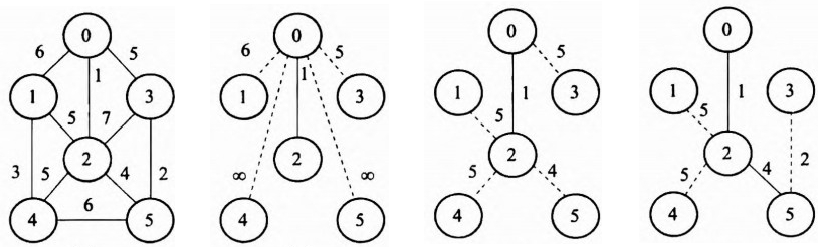
Prim 算法的原理是，从指定顶点开始，将它加入集合中。然后将集合内的顶点与集 合外的顶点所构成的所有边中选取权值最小的一条边作为生成树的边，并将集合外的那个 顶点加入到集合中，表示该顶点已连通。再用集合内的顶点与集合外的顶点构成的边中找 最小的边，并将相应的顶点加入集合中。如此下去，直到全部顶点都加入到集合中，即得 最小生成树。图8-12 (a) 为一个无向图，图上包含了各个边的权重。

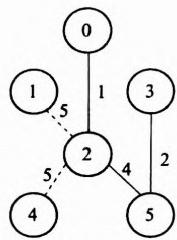
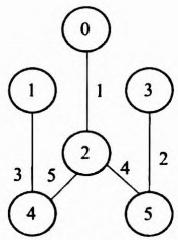
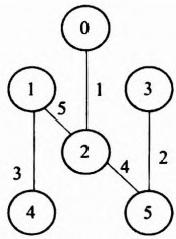
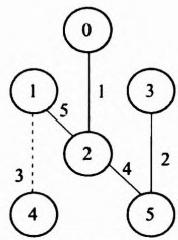
Prim 算法的各个步骤描述如表8-4所示。

图8-12中还显示了另外一个最小生成树，因为在第四步中，可以选② →④边。

**表8-4** **Prim算法的步骤**

|  |  |
| --- | --- |
| **步骤** | **描述** |
| 1 | 从◎开始，◎进集合。◎与集合外所有顶点能构成的边中找最小权值的一条边。 ◎→①的权值为6  ◎→②的权值为1,所以取◎→②边  ◎→③的权值为5 |
| 2 | ②进集合，寻找◎,②与①,③,④,⑤,⑥构成的最小边 ◎→③的权值为5  ②→①的权值为5  ②→④的权值为5  ②→⑤的权值为4.所以取②→⑤边 |
| 3 | ⑤进集合，寻找◎,②,⑤与①,③,④构成的最小边  ②→①的权值为5  ②→④的权值为5  ⑤→③的权值为2,所以取⑤→③边 |
| 4 | ③进集合，寻找◎,②,⑤,③与①,④构成的最小边  ②→①的权值为5,所以取②→①边  ②→④的权值为5 |
| 5 | 进集合，寻找◎,②,⑤,③,①与④构成的最小边 ①→④的权值为3,所以取①→④边 |

(a) (b) (c) (d)



(e) (f) (g) (h)

**图8-12** **最小生成树构造过程**

**4. 网络主干** **(Network Backbone)**

在网络的边数很多的情况下，我们很难对网络进行观察。对这些边进行裁剪，保留重 要的边，删除一些不重要的边，对网络进行精简，有利于对网络进行观察和分析。比如， 在引文网络 (Paper Citation Network) 中，经过精简，有利于我们观察到科学思想的大致 传播路径。在这里我们假设论文 B引用了论文A, 表示科学思想从A 传播到了B。

(1)Pathfinder Network Scaling(PFNet)。Pathfinder Network Scaling是认知心理 学家 (Cognitive Psychologists)Schvaneveldt等，于1990年提出的一种结构建模 (Struc- turing Modeling) 技术。这项技术用于刻画概念之间重要的、突出的关系。关系的强度就 是概念的相似度，是认知心理学家给出的一个相似度评分。

该算法的输入是一个相似矩阵，算法对网络进行裁剪，去除多余的连接/路径。它运 用欧式空间的三角不等式原理 (Triangular Inequality) 进行决策，即两个顶点的距离小于 或者等于连接这两个顶点但是通过第三个顶点的其他路径。知识图谱分析软件Sci2 已经集 成了PFNet 算法。PFNet 算法的具体步骤，请参考相关资料。

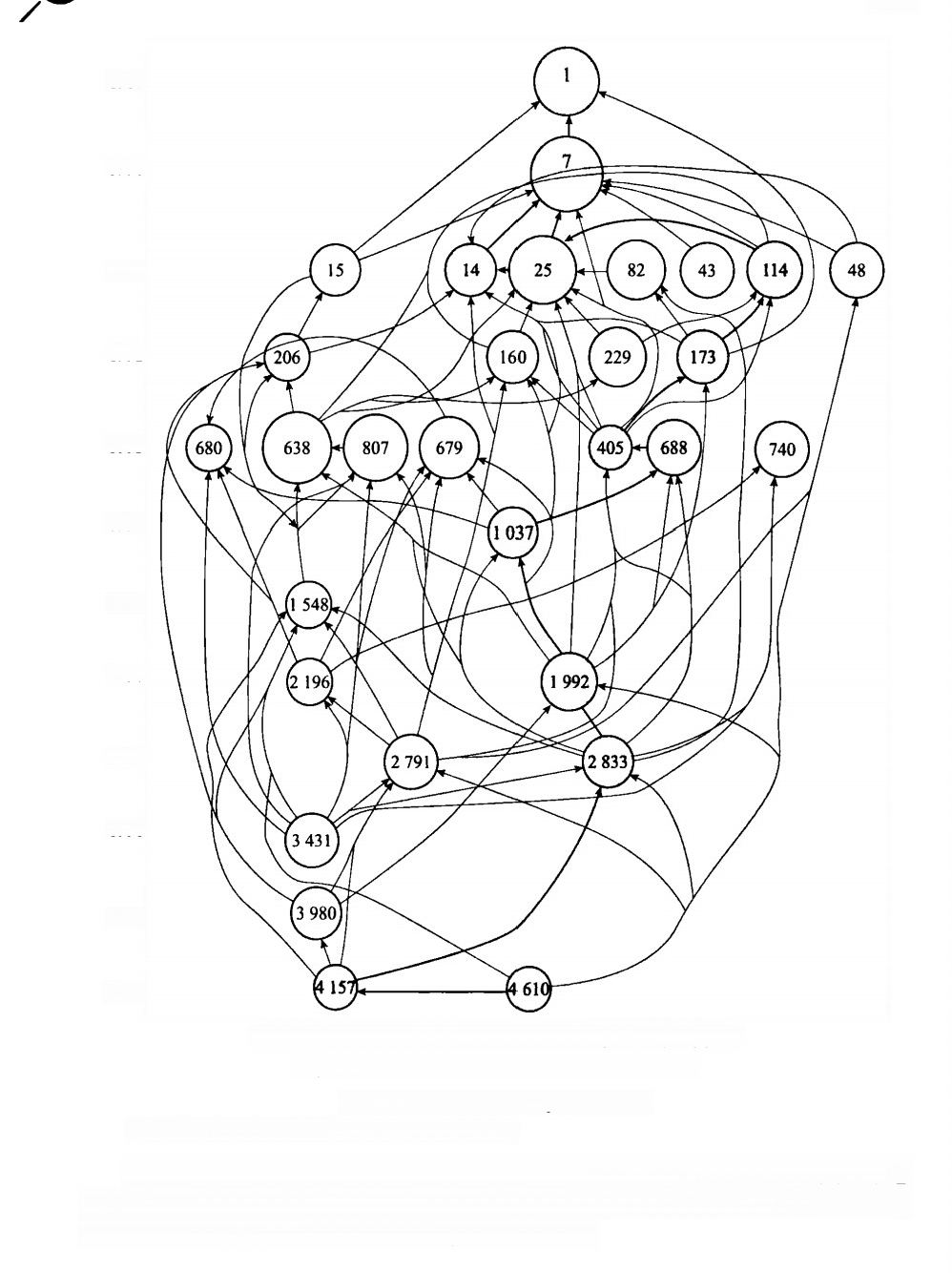
PFNet 算法的目标是寻找最短路径，其时间复杂度较高，为O(n⁴)(n 是边的数量),处 理大型网络时其运行时间过长。基于最小生成树的Pathfinder算法 (MST Based Pathfind- er Algorithm), 把时间复杂度降低到O(n²·logn) 。 该算法的核心思想建立在如下事实之 上，就是一个网络的所有最小生成树的并集 (Union). 等价于 Pathfinder 算法生成的 网络。

(2)主路径分析方法 (Main Path Analysis,MPA) 。1989 年 ，Hummon 首次提出了 主路径分析方法，用于对 DNA 学科的引文网络进行分析，从中提取出一些起重要作用的 节点。这些节点之间的连接关系代表了DNA 学科发展过程中主要的思想传播路径。

为了进行主路径分析，需要考察顶点间连接的重要程度， 一般用搜索路径数 (Search Path Count,SPC) 来衡量。一个连接的SPC 值，可以由网络中从所有起点出发到所有终 点结束所经过的所有路径中穿过此连接的次数来衡量。被穿过的次数越多，即SPC 值越 大，这个连接也就越重要。除了SPC, 还有节点对投影数 (Node Pair Projection Count, NNPC) 、搜索路径连接数 (Search Path Link Count)、搜索路径节点对 (Search Path Node Pair,SPNP) 等衡量重要性的指标。

在引文网络的主路径分析中，需要计算一篇文章与其他文章连接的重要程度，称为遍 历值或者遍历权重 (Traversal Count,Traversal Weight)。首先识别出源节点(数据集中 没有引文的文章，图8-13中的节点1)和汇节点(数据集中没有被其他文章引用的文章， 图8-13中的节点4610)之间的所有路径。然后计算包含特定引文 (A 引 用B) 的路径 数。最后，用包含特定引文的路径数，除以从源节点到汇节点的总路径数，这个比率就是 遍历权重。

有了引文的遍历权重，就可以抽取出这些连接中具有最高遍历权重的路径，形成主 路径。主路径表达了某个学科领域的主要知识传播脉络，通过主路径分析，可以展示出 某个学科领域的知识的集成、分化等演进模式。 HistCite,Pajek等主流社交网络分析软 件集成了主路径分析方法。图8- 13是利用 HistCite 进 行 Main Path Analysis 分析的 实例。

1987

1990

1991

1992

1993

1994

1995

1996

1997

**1998**

**1999**

2000

(1987—2000年间标题中包含 Fullerene 的文章里引用量最大的

Top30 个文档集上的 Main Path Analysis分析结果)

**图8** **-** **13** **Main** **Path** **Analysis分析结果**

资料来源：<http://leydesdorff.net/histcite/histcite.pdf.>

PFNet 算法生成的网络过于复杂，我们可以利用MPA 方法，对PFNet 生成的网络进 行简化，以便观察。除了上述算法，寻找网络主干 (Backbone of a Network) 的算法，还 有最大流 (Maximum Flow) 算法等，请参考相关资料。

**8.3.5** **凝聚子群与社区检测**

在网络中，某些行动者之间的联系特别紧密，以至于形成一些团体。这些团体在社交 网络分析中称为凝聚子群 (Cohesive Subgroup), 有些学者则形象地称之为“小团体”。下 面，我们来了解不同类型的凝聚子群。

(1)派系 (Clique) 。 派系是最大完备子图 (Maximally Complete Subgraph)。在 一 个 无向图中， 一个派系：(a) 最少包含三个顶点；(b) 派系是完备的，派系中任意两个顶点 之间存在连接。派系是最大的 (Maximum Clique), 向这个子图再增加任何顶点都破坏其 完备的性质。

给定无向图G=(V,E), 其 中V 是非空顶点集， E 是边集，是V 中元素构成的无序二 元组的集合。如果UCV 且对任意u,v ∈U, 有 (u,v) 不属于E, 则 称U 是G 的空子图。G 的空子图U 是G 的独立集，当且仅当U 不包含在G 的更大的空子图中。G 的最大独立集是 G 中所含顶点数最多的独立集 (Maximum Independent Set)。

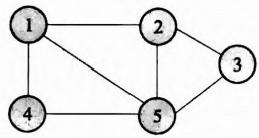
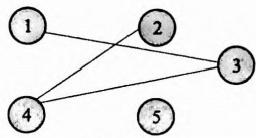
对于任一无向图G=(V,E), 其 补 图G′=(V′,E') 定义为：V′=V, 且 (u,v)∈E′ 当 且仅当(u,v)∉E 。 如 果U 是 G 的完全子图，则它也是G′ 的空子图 .反之亦然。因此，G 的 团 ( 派 系 ) 与G′ 的独立集之间，存在一一对应的关系。特殊地，U 是G 的最大团，当且 仅当U 是G′的最大独立集 (Maximum Independent Set)。

如图8- 14所示，给定无向图G={V,E}, 其 中V={1,2,3,4,5},E={(1,

2),(1,4),(1,5),(2,3),(2,5),(3,5),(4,5)}。右侧子图是无向图G 的 补 图 G'。

根据最大团 (Maximum Clique) 定 义 ， 子 集 { 1 , 2 } 是 图G 的 一 个大小为2的完全 子图，但不是一个最大团，因为它包含于G 的更大的完全子图{1,2,5}之中。{1,2, 5 } 是G 的 一 个最大团，{1,4,5}和{2,3,5}也是G 的最大团。

根据最大独立集定义，{2,4}是G 的一个空子图，同时也是G 的一个最大独立集。 虽 然 { 1 , 2 } 是G′ 的空子图，但它不是G′ 的最大独立集，因为它包含在G′ 的空子图{1, 2 , 5 } 中 。 { 1 , 2 , 5 } 是G′的 最 大 独 立 集 ， { 1 , 4 , 5 } 和 { 2 , 3 , 5 } 也 是G′的最大独 立集。



G *G′*

**图8-14** **最大派系、最大独立集**

这里，我们介绍图的K 着色问题。给定一个无向连通图G 和 K 种不同的颜色，用这 些颜色为图G 中的各个顶点着色，每个顶点一个颜色。是否有办法保证G 中每条边连接 的两个顶点着不同的颜色，这是图的K 可着色判定问题。

如果一个图最少需要 K 种颜色才能使图中每条边连接的两个顶点着不同的颜色，那 么称这个数K 为该图的色数。求一个图的色数K 的问题，称为图的K 可着色问题。

著名的四色定理指出，平面或者球面上，任何地图的所有区域都至多可用4种颜色进

行着色，并且保证有公共边界的相邻区域没有相同的颜色。这个问题可以进行如下转换， 将地图的每个区域变成一个顶点，如果两个区域相邻，则相应的顶点之间用一条边连接起 来。已经证明用5种颜色足以对任何一幅地图着色，但是多年来一直未能够证明最多用4 种颜色能够对地图进行着色。1976年，这个问题由爱普尔、黑肯和考西利依靠电子计算 机的帮助得以解决。①他们证明了4种颜色足以对任何地图着色。

(2)n- 派系 (n-Clique) 。 无向图的一个n- 派系是这样的一个最大子图 (Maximal Sub- graph), 每对顶点间，通过小于等于n 的路径连接起来。换个说法，在一个总图中的一个 导出子图 (Induced Subgraph) 中，它的任意两个顶点之间在总图中的最短路径，最大长 度不超过 n 。π-派系可以用Bron &Kerbosch(1973) 算法的一个改进版来寻找。

(3)n 宗派 (n-Clan) 。 一 个 n-宗派同时是一个n-派系，并且其半径在子图中小于等 于n。也就是在子图中，其中任何两点之间的最短路径长度都不超过n。

(4)k- 丛 (k-Plex)。k 丛是满足下列条件的一个最大子图 (Maximal Subgraph),每 个顶点都至少与除了k 个顶点之外的其他点直接相连。在无向图中，当一个凝聚子群的规 模为n 时，其中每个点至少都与该凝聚子群中n—k 个顶点有直接联系，即每个点的度数 都至少为n—k, 那么该凝聚子群为一个k- 丛。可以通过深度优先搜索算法 (Depth First Search) 寻找k- 丛。

(5)k- 核 (k-Core) 。k- 核是最大的一组行动者 (Maximal Group of Actor), 每个行动 者连接到该组行动者中的其他k 个行动者。

**社区检测** **(Comnunity** **Detection)**

对于大型的社交网络来讲(往往包含超过百万的顶点),在单个节点的粒度上进行可 视化和分析是相当困难的。为了加深对社交网络的理解，一种办法是社区检测，这是一种 对网络进行粗粒度刻画和描述的方法。

最经典的社区检测算法是Louvain 算法。该方法是一个非重叠社区检测算法，它具有 较快的执行效率，可以分析大型的网络。典型的包含200万个顶点的网络，通过该算法寻 找社区，需要的时间为2分钟左右。由于算法发现的社区特有的层次性结构 (Hierarchical Structure) 特点，方便用户在不同的分辨率下观察网络和社区。不管网络是无向的还是有 向的，只要顶点间的关系是一样的。算法检测的结果就是一样的。该算法可以运行在有权 重的图上，如果边没有权重，那么所有的边被赋予权重1。

该算法的目标是最大化一个模块度函数 (Modularity Function)。模块度是评估一个 社交网络的结构强度的方法，它的含义是社区内节点连接的边数与随机情况下的边数之 差.最早由 Mark Newman 提出，它的取值范围是[-1/2,1]。

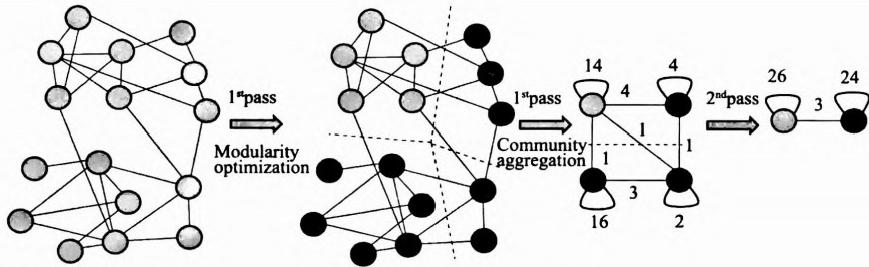
该算法的基本流程是：(1)刚开始时，所有的顶点都是一个小小的类簇。(2) Phase 1: 以局部方式，优化模块度函数，将每个顶点归到“最好”的类簇中，直到所有的顶点 所属的类簇不再变化为止。(3) Phase 2: 对网络图进行缩减 (Reduce), 也就是把一个类 簇中的所有顶点聚集抽象为一个顶点，重建一个网络，其中的每个顶点对应一个社区。看 抽象以后的网络图是否还有优化的可能性，如果有，则迭代执行 (Iteratively) 上述(2)、

(3)步骤。图8- 15展示了Louvain 算法的主要步骤。

每一轮迭代，我们获得对网络的一个划分 (Partitioning) 。 第一轮迭代获得的划分， 每个类簇一般都比较小，也就是每个社区包含较少的顶点。其后的迭代，我们就能够通过



①<https://en.wikipedia.org/wiki/Four_color_theorem.>



**图8-15** **Louvain 算法原理**

资料来源：<https://arxiv.org/pdf/0803.0476.pdf.>

聚集机制 (Aggregation Mechanism), 找到越来越大的社区。算法的执行过程产生了大社 区包含小社区的层次性结构 (Hierarchical Decomposition of the Network)。算法的输出是 给每个顶点增加一个标记 (Annotation), 标记的形式为 “community\_level\_x”, 其 中x 为某个层次网络社区的社区 ID。

从算法的原理可以看出，Louvain 算法不能保证获得全局最大的模块度 (Global Max- imum of Modularity),但是对该算法的实际测试结果表明，算法具有良好的准确度，算法 的社区检测结果，其模块度接近全局最优。

除了Louvain 社区检测算法，Louvain 多层次精化社区检测算法 (Louvain Multilevel Refinement Community Detection Algorithm) 和 SLM 社区检测算法 (Smart Local Mov- ing Community Detection Algorithm) 能够获得更高的模块度，但是这些算法需要更多的 执行时间，没有 Louvain 算法这么快。

在现实的社交网络中， 一个顶点可能属于多个团体。支持重叠社区 (Overlapping Community) 发现的算法有CFinder(<http://hal.elte.hu/cfinder/wiki/>) 等。

在社交网络分析中还需要了解，随着时间变化，团体是如何创建和发展的 (Dynamic Community Finding)。我们可以为社交网络建立历史快照 (Snapshot), 并且对每个快照 应用社区检测算法。然后将新生成“后续团体”与过去发现的团体进行匹配，对社区的变 化进行考察，这些变化包括产生 (Birth) 、 消 亡 (Death) 、 扩 大 (Expansion) 、 缩 小 (Contraction) 、 合并 (Merging) 、 分解 (Splitting) 等 。

**8.3.6** **链路预测、信息扩散与影响力分析**

**1. 链路预测** **(Link Prediction)**

链路预测 (Link Prediction) 是社交网络分析中一个重要而有趣的基本问题。它找出 网络中“不存在但是应该存在的边”(有可能被我们观察漏掉的),或者“现在不存在但是 将来可能存在的边”(未来会出现的一些链路)。

具体到社交网络，对于前一种情况，比如两个人实际上认识，但是在社交网络网站上 没有互相加好友。对于后一种情况，比如现在两个人并不认识，但是具有成为好友的潜 质，因为两个人有共同的爱好，有较多的共同好友等。链路预测的主要算法介绍如下。

(1)基于相似度的算法(Similarity based Algorithm)。对于每对顶点x 和y, 计算一



个得分S, 表示x 和y 之间的相似度。所有没有观测到的连接 (Non-observed Link), 根据其得分进行排序。相似度更高的顶点之间的连接，极有可能是实际存在的 (Higher Existence Likelihood)。

我们可以基于顶点的属性 (Attributes of Node) 定义顶点的相似度。如果两个顶点之 间有很多的共同特征 (Common Feature),那么这两个顶点被认为是相似的。但是，顶点 的属性一般是隐藏的(不易观测到)。我们还可以根据网络结构 (Network Structure) 定 义顶点相似度，称为结构相似度(Structural Similarity)。

人们提出了总共20余个相似度指标，可以分为局部指标 (Local Index)、全局指标 (Global Index) 和半局部指标 (Quasi-local Index) 三类。其中，共同邻居数量 (Common

Neighbors,CN) 是最常用的局部指标。但是，在实际应用中，其他指标比如 Jaccard In- dex 的效果有时比CN 指标要好。其计算公式如下，P(x) 表示x 的邻居节点：



下面我们介绍一个全局相似度指标(其他指标的含义和计算方法，请参考相关资料)。 Katz Index是一个常用的全局相似度指标，它的计算公式为：



paths,<> 表示连接顶点x 和顶点y 的所有长度为l 的路径，β是一个参数，控制不同 长度路径的权重。这个公式的含义是，全局相似度依赖于顶点i 和顶点j 之间的所有路径， 更短的路径的影响更大。

(2)最大似然方法 (Maximum Likelihood Method)。基于最大似然估计的算法，假设 网络结构是符合某些组织原则 (Organizing Principle) 的，这些原则体现为一系列的规则 和参数。这些参数可以通过对观察到的网络的最大似然估计 (Maximizing the Likelihood of the Observed Structure) 而获得。根据估计这些规则和参数，可以对未观测到的连接 (Non-observed Link) 的可能性 (Likelihood) 进行计算。

这类方法比较耗时，同时连接预测的准确度不是很高，但是它能够对网络是如何组织 的提供一些有价值的洞见 (Insight) 。 主要的网络模型有层次结构模型 (Hierarchical Structure Model) 和随机块模型 (Stochastic Block Model) 等。根据人们的实际经验，很 多的实际网络具有层次性的组织结构，顶点划分成群组 (Group), 群组又继续划分成不同 的子群组 (Sub Group) 等，层次结构模型可以对其进行刻画。随机块模型，也是最通用 的网络模型之一，顶点被划分成群组 (Group), 两个顶点连接的概率 (Connect Probabili- ty) 唯一依赖于它们隶属于哪个群组。这个模型可以较好地捕抓到网络里的社团结构 (Community Structure)、角色关系 (Role-to-Role Connection) 等决定顶点间是否有连接 的重要因素。这个模型能够刻画群组的成员资格 (Group Membership) 对于两个顶点是 如何交互地发挥作用的情况。

(3)概率模型。概率模型 (Probabilistic Model) 试图从观察到的网络 (Observed Network) 抽象出其结构，然后用学习到的模型预测缺失的连接。给定一个目标网络 G=(V,E), 概率模型将优化一个目标函数 (Target Function),从而建立一个模型。这 个模型和观察到的目标网络达到最佳的拟合 (Best fit)。这个模型包含若干参数，表示为 Θ。顶点i 和顶点j 之间是否存在连接，其概率通过条件概率 P(A;,=1|0) 来估计。主流



的概率模型包括概率关系模型 (Probabilistic Relational Model,PRM)、概率实体关系模

型 (Probabilistic Entity Relationship Model,PERM) 以及随机关系模型 (Stochastic Re- lational Model,SRM) 等。这些模型的具体细节，请参考相关资料。

链路预测有广泛的应用。比如，在生物学领域，我们观察到的网络 (Observed Net- work) 只是真实网络中的一部分，需要通过实验来确定其他的部分，也就是其他的连接， 这个问题称为网络的重建 (Reconstruction) 。 网络的重建，不仅要考虑缺失的连接 (Missing Link) 的重建，而且需要对伪造或者错误的连接 (Spurious Link) 进行鉴别，把 它们剔除。

在社交网络领域，可以通过链路预测来进行好友推荐。准确的链路预测算法，还可以 对网络演化模型进行评估。人们已经提出了大量的模型，试图解释不同网络的演化机制 (Evolving Mechanism),到底哪个模型更加接近实际情况呢?网络的演化模型 (Evolving Model) 可以映射到一个链路预测算法，也就是网络连接是如何不断建立、不断消失的。 通过度量链路预测的准确度指标 (Performance Metrics for Link Prediction), 我们可以比 较不同模型的准确度。

**2. 扩散模型** **(Diffusion Model)、影响力分析与影响力最大化** **(Influence Analysis & Maximization)**

(1)扩散模型 (Diffusion Model)。观 点 (Opinion) 、 想 法 (Idea) 、 信息 (Informa- tion) 、 创新技术和思想 (Innovation) 等，在社交网络的成员间传播。社交网络成为影响 力 (Influence) 在其成员间传播的媒介，其中一个重要的应用是口碑和病毒式市场营销 (Word of Mouth and Viral Marketing) 。68%的消费者在购买家用电子产品之前都会咨询 朋友或者家人，在进行市场营销时，我们就可以把重点放在寻找这样的目标用户上，他们 更有可能激发 (Invoke) 口碑的传播 (Word of Mouth Diffusion), 我们可以使用有限的 预算达到较好的营销效果。

观点、想法、信息、创新技术和思想等在社交网络成员之间的传播过程，需要用适当 的扩散模型来刻画。扩散模型也称为传播模型 (Propagation Model)。主要的模型有 IC 模 型 (Independent Cascade Model) 和 LT 模型 (Linear Threshold Model)。

在 Independent Cascade Model里，每条边(u,v) 都附带一个概率p(u,v), 表示 u 对v 的影响的概率。时间按照离散的步骤递进，在时间步t, 由时间步t-1 已经被激活的顶 点，试图激活未激活的邻居节点，激活成功的概率即p(u,v)。

在 Linear Threshold Model里，每条边 (u,v) 都附带一个权重b(u,v), 并且保证进入 每个顶点的权重之和≤1。时间按照离散的步骤递进，每个顶点选定一个随机的阈值 (Random Threshold)θ,∈[0,1]. 当从激活的邻居顶点，进人一个顶点的所有权重之和 (Sum of Incoming Weights) 达到θ,那么该顶点被激活。

上述模型刻画了社会影响力 (Social Influence) 的传播。社会影响力指的是一个顶点 (比如顶点A) 施加于其他顶点(比如顶点B) 的一种力量。这个力量将改变顶点B 的 行 为或者观点 (Behavior/Opinion) 。 影响力体现了一种因果关系 (Causal)。

对于实际的社交网络，我们掌握一部分历史信息，这些信息描述了影响力传播关系。 比如在Twitter 网络上，A 关注 B. 那么表示影响力从B 传播到了A 。为了使用上述模型 表达影响力传播过程，需要确定模型的参数。我们可以使用EM 算法 (Expectation Maxi- mization Method), 从实际数据中学习到模型的参数。

(**2)影响力最大化。影响力最大化** **(Influence** **Maximization)** **指的是给定一个社交网**



络G 以及一个扩散模型M 。 要寻找出Top-K 个初始的顶点，作为种子集合S 。 目的是最 大化影响力的传播和扩散oM(S)。oM(S) 表示以S 作为初始的激活顶点集 (Initial Set of Active Nodes) 时，影响力传播或者扩散的预期范围大小 (Expected Size)。一般来讲，影 响力最大化是有预算限制的 (Budget Limit)。

对应到病毒式市场营销场合，影响力最大化就是要找到具有影响力的人，构成一个种 子集合，针对性地对他们进行市场营销，通过他们对其他人的影响，提高和扩大产品知名 度的扩散速度 (Speed) 和范围 (Spread) 。 影响力最大化问题是NP 困难的 (NP-Hard),

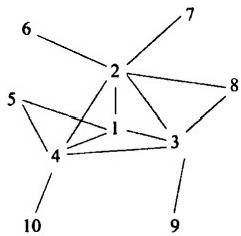
也就是无法在多项式的时间复杂度里给予解决。但是我们可以使用贪心算法解决这个问 题，其原理是，寻找一个顶点，该顶点的选择依据是，如果把它加入初始顶点集，oM 相对 于把其他顶点加入顶点集更大。寻找下一个顶点，直至找到k 个顶点为止，具体如下。

|  |
| --- |
| Greedy Algorithm for Influence Maximization Input:G,k,om  Output:Seed set S  1,S←  2,while |S|<k do  3,select u =argmaxuevn.(om(SU{w})—om(S))  4,S←SU {w} |

**8.3.7** **核心-边缘分析**

核心-边缘 (Core-Periphery) 结构分析的目的是研究社交网络中。哪些顶点处于核心 地位 (In the Core of the Network), 哪些顶点处于边缘地位 (On the Periphery)。核心- 边缘结构分析有广泛的应用.可用于科学引文关系网络、组织关系网络等多种类型的网络 中体现出来的核心-边缘结构。

图8-16展示了一个网络(无向图),直观来看它具有典型的核心-边缘结构，其中顶 点1,2.3,4构成了核心。其他顶点则属于边缘顶点。网络的一些顶点(行动者 (Ac- tor)) 之间具有高密度的联系 (Tie), 称为网络的核心 (Core) 。 另外一些顶点之间具有低 密度的联系，称为网络的边缘 (Periphery) 。 边缘顶点和核心顶点之间可能有连接关系。



**图8-16** **具有核心-边缘结构的一个网络**

人们提出了两类核心-边缘结构 (Core Periphery Structure) 模型，分别是离散模型 (Discrete Model) 和连续模型 (Continuous Model)。其中，在离散的核心-边缘结构模型

中，根据核心成员和边缘成员之间关系的有无以及关系的紧密程度，划分为如下几类：

(1)核心-边缘全关联模型。网络中的所有顶点分为两组，其中一组的成员之间联系紧密， 可以看成一个凝聚子群(核心),另外一组的成员之间没有任何联系，但是该组成员与核 心组的所有成员之间都存在关系。(2)核心-边缘局部关联模型。网络中的所有节点分为 两组，其中一组的成员之间联系紧密，可以看成一个凝聚子群(核心),另外一组的成员 之间没有任何联系，但是它们同核心组的部分成员之间存在联系。(3)核心-边缘无关模 型。网络中的所有节点分为两组，其中一组的成员之间联系紧密，可以看成一个凝聚子群 (核心),另外一组的成员之间没有任何联系，并且同核心组成员之间也没有联系。(4)核 心-边缘关系缺失模型。网络中的所有节点分为两组，其中一组成员之间的密度达到最大 值，可以看成一个凝聚子群(核心),另外一组成员之间的密度达到最小值，但是并不考 虑这两组成员之间关系密度而是把它看作缺失值。

实际的网络并未表现出如图8-16那样的理想化的核心-边缘结构，但是我们可以针 对一个网络找到一组顶点，它们之间具有较高密度的连接，另外一组顶点具有较低密度的 连接。到底实际网络中多大程度上接近理想情况，用如下的度量值 (Measure) 来进行描 述，也称为拟合函数 (Fit Function)。





式中，a; 表示顶点i 和顶点j 之间是否具有连接；c; 表示目前某个顶点隶属于Core 还是Pe- riphery;δ; (构成Pattern Matrix) 表示在理想化的核心-边缘结构中，顶点i 和顶点j 之间 是否具有连接。当且仅当矩阵A(matrix of a;) 和矩阵△ (Matrix of 8;)相等时，p 具有最 大值，这时候网络具有理想化的完美的核心-边缘结构。换句话说，p 有多大，意味着网络 多大程度上接近理想化的、完美的核心-边缘结构。

基于上述优化目标(最大化p), 我们可以使用组合优化技术，比如模拟退火算法 (Simulating Annealing) 、Tabu 搜索算法 (Tabu Search)、遗传算法 (Genetic Algorithm) 等，寻找网络中的核心-边缘结构。很多社交网络软件，包括UCINET 等，已经集成了寻 找核心-边缘结构的算法。

8.3.8 位置和角色、子图查询、网络模体

1. 位置和角色

在社交网络中，位置和角色是两个重要的概念。在社交网络中，顶点的社会位置 (Social Position) 或社会角色 (Social Role) 定义了顶点的类别，取决于顶点和其他类别 的顶点之间的关系。换句话说，结构分析者 (Structural Analyst) 通过行动者 (Actor) 之间的关系，来研究不同类型的社会位置和社会角色，而不是通过行动者本身的属性来研 究。当我们说两个行动者具有相同的位置和角色时，我们的意思是他们和其他行动者具有 相同的关系模式。

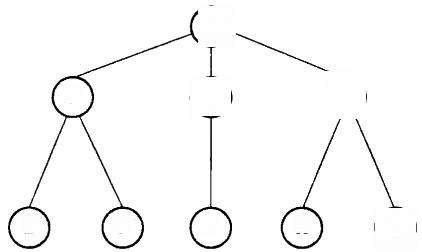
比如，对于“丈夫”(Husband) 这样的社会角色的定义，可以把它看作和其他的社 会类别“妻子”(Wife) 和“孩子”(Child) 等的顶点具有某种关系的一种顶点类别。

在网络分析中，一般通过“等价类”(Equivalence Class) 的概念来定义两个顶点(或



者两个网络的精细结构)的相似性。在顶点属于何种位置或者角色方面，顶点之间是否等 价取决于它们和其他顶点的关系。

有三类等价性定义，分别是结构等价性、自同构等价性和正则等价性，这三类等价性 有助于我们进一步理解顶点的位置和角色，通过图8-17的网络实例加以说明(这个网络 由 Wasserman 和 Faust 提出)。





B

E F G H ①



(用于说明结构等价性、自同构等价性和正则等价性的Wasserman-Faust网络)

图8-17 Wasserman-Faust 网络

资料来源： <http://faculty.ucr.edu/>～hanneman/nettext/C12\_Equivalence.html#approach.

(1)结构等价性 (Structural Equivalence)。

如果两个顶点和其他所有顶点具有相同的关系模式，两个顶点是结构等价的。比如在 图8- 17中有7个结构等价类别 (Structural Equivalence Class)。

①A 构成一个独立的类别，因为没有其他的顶点具有和它一样的和其他顶点的连接 (A 连接到B,C,D)。

②同样道理， B,C,D 也分别构成独立的类别。

③E 和 F 构成一个结构等价类，因为它们只有一个连接，指向B, 于 是E 和 F 具有相 同的和其他顶点的连接，换句话说，它们是结构等价的。

④同理，G 构成一个独立的类别，它只和C 连接，没有其他节点和C 之间有这样的连 接模式。H 和 I 构成一个结构等价类。

结构等价性是一个强大的概念，因为具有结构等价性的顶点处于相同的位置，它们之 间可以完全互相替代 (Substitutable)。

(2)自同构等价性 (Automorphic Equivalence)。

还有比结构等价性不那么严格 (Less Strict) 的等价性定义，用于定义行动者之间的 相似性或者等价性。

假设图8-17描述的是一个快餐公司的特许经营网络， A 是总部 (Headquarter),B,

C.D 是三个分店的经理 (Manager),E,F,G,H,I 分别是各个分店的员工 (Worker)。

虽 然B 和D 不是结构等价的 (Structurally Equivalent)——因为他们的老板是同一个 而员工不一样，但我们还是感受到他们某种另外的等价性 (Equivalence)。B 和 D 都是向 一个老板报告，他们都有两个员工。他们是不同的人，但是他们看起来具有一定的等价 性。如果我们在这个网络中交换这两个经理，并且交换其员工(四个员工),网络中所有 行动者之间的距离和原来的网络是一样的。

在这里，B 和 D 所表现出来的等价性使得他们属于一个自同构等价性类别 (Auto- morphic Equivalence Class)。在图8 - 17中总共有5个这样的类别，分别是{A},{B,

**D},{C},{E,F,H,I} 和{G}。**

**(3)正则等价性。**

正则等价性 (Regular Equivalence) 描述了构成社会组织的基本模块 (Basic Building

**Block)** **的社会角色** **(Social** **Role)。具有正则等价性的行动者，具有和其他行动者集合** **(这些集合也是正则等价类)的某些成员相同类型的关系。正则等价性体现了从社会学视** **角** **(Sociological Perspective) 观察到的制度化** **(Institutionalized) 的社会结构** **(Struc** - ture) 里不同的社会角色。

在图8-17中，有三个正则等价性类 (Regular Equivalence Classes),分 别 是 {A}, {B,C,D},{E,F,G,H,I}。

在图8-17中最底层的五个行动者E,F,G,H 和1,他们是正则等价的，因为他们 具有和其他类别的行动者相同的关系模式 (Pattern of Tie),即 (a) 他们和第一类行动者 {A} 没有关系；(b) 他们和第二类行动者{B,C,D} 的某个行动者有一个连接关系。

B,C 和 D 构成一个正则等价类，因为他们都 (a) 和第一类行动者{A} 有一个连接 关系；(b) 他们每个都和第三类行动者的成员有一个连接关系。虽然 B 和 D 实际上和第 三类行动者的成员有两个连接关系，C 只和第三类行动者的成员有一个连接关系，但是这 无关紧要，他们都和第三类行动者的成员有至少一个连接关系。

A 是一个行动者，构成一个等价类，因为 (a) 他和第二类行动者的至少一个成员有 一个连接关系；(b) 他和第三类行动者的任何成员都没有任何连接关系。

**2.子图查询**

子图查询 (Sub Graph Search) 可以给出如下的形式化定义。给定一个模式图 (Pat- tern Graph)G1和一个数据图 (Data Graph)G2, 判断G1 是否匹配G2, 或者从G2 中找 出所有跟G1 匹配的子图。这里的图是由顶点集和边集构成的。模式图G1 一般较小，包 含几个或者几十个顶点。数据图G2 一般较大，通常包含上亿级别的顶点和边。

子图查询或者图匹配查询有广泛的应用，包括社会关系查找、角色分析、专家推荐 等。(1)社会关系查找，我们可以查找给定的A 和 B 是不是远房亲戚，查找A 和 B 是不 是三代以内的近亲，查找A 的所有三代以内具有血缘关系的亲属等，这些查询可以通过带 有边属性约束的可达性查询来完成。(2)角色分析，当经济不景气时，某个公司需要裁 员，这时公司需要了解每个员工对公司的重要程度，哪些人的工作是可以互相替换的，也 就是分析角色的等效性，然后把冗余的人裁掉。(3)专家推荐，比如一个理想的软件开发 团队包括项目管理员、业务需求分析师、软件架构师、用户界面设计师、软件开发员以及 软件测试员等。我们需要从专家社交网络中找到合适的专家来组建这样的团队。

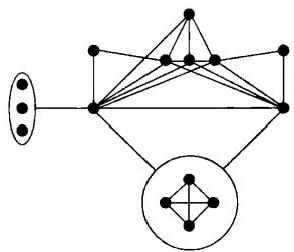
为了实现图匹配查询，需要模式图预处理、数据图预处理、索引等相关技术。(1)模 式图预处理：对模式图进行预处理可以提高查询的效率。常见的模式图预处理有模式图最 小化和模式图相似变换等。前者是保证查询结果不变的前提下，对模式图的冗余顶点和边 进行最大化的剔除。后者是将模式图简化为相似的模式图，并尽量保证查询结果的正确 性。(2)数据图预处理：主要技术包括取样、压缩和划分。数据图取样是对数据图进行取 样处理，并在取样后的数据图上执行匹配任务。需要在允许一定精度损失的情况下，通过 筛选，减少需要处理的数据图的大小。数据图压缩是通过减小数据图的规模来提高查询效 率。根据具体的应用(可达性查询、邻接关系查询等)所需的信息，对原始数据图进行处 理，运用信息编码等技术，对数据进行压缩。数据图划分则将一个大的数据图划分成若干 小图，并将其分布在若干计算节点上，通过并行处理，提高匹配的速度。(3)在数据图规

模较大的情况下，在其上建立索引①,可以有效提高图匹配查询，但是需要在索引大小、 索引构建时间、查询时间的加速效果之间做出折中和平衡。

**3.** **网络模体**

网络模体 (Network Motif) 是指网络中若干顶点之间的相互连接模式 (Interconnec- tion Pattern), 这种模式在实际的复杂网络中出现的比随机网络 (Randomized Network) 更加频繁。网络模体被认为是构成复杂网络的基本模块 (Building Block)。人们在生物化 学 (Biochemistry) 、 神经生物学 (Neurobiology) 、 生态学 (Ecology) 和工程学 (Engi- neering) 等领域，都从大型网络中发现各类网络模体。

图8- 18是酿酒酵母 (Saccharomyces Cerevisiae) 的蛋白质交互网络 (Protein-Pro- tein Interaction Network) 中分析出来的某个常见的网络模体，它包含15个顶点、34条 边。注意，从一组顶点指向某个顶点(比如n) 的一条边，表示该组内的所有顶点都和顶 点 n 有连接。在该蛋白质交互网络中，这个网络模体出现了27720次之多。



**图8** **-** **18** **蛋白质交互网络中的网络模体**

资料来源：<http://compbio.mit.edu/publications/13_Grochow_LectureNotesBioinformatics_07.pdf.>

**8.4** **软** **件**

**8.4.1 Gephi**

Gephi 是一款开源的免费社交网络分析(图分析)软件，它可以运行在 Windows, MacOS X,Linux等平台上。 Gephi 应用于探索式数据分析 (Exploratory Data Analy- sis) 、 链接分析 (Link Analysis)、社交网络分析 (Social Network Analysis) 以及生物网 络分析 (Biology Network Analysis) 等领域。

探索式数据分析通过实时地操控 (Real Time Manipulation) 可视化的网络，依赖于 人们的直觉 (Intuition), 获得对网络结构的直观认识，发现潜藏的模式。Gephi 实现了实 时的可视化功能，可以支持100000个节点、1000000条边的网络实时可视化。

在网络布局方面 (Layout),Gephi 支持最新 (State of the Art) 的布局算法，兼顾效



①关于具体索引技术的细节信息，请参考相关资料。

率和质量，其可视化结果可以导出到PDF,SVG,PNG 等文件格式，有利于科研工作者 基于这些文件制作高质量的论文插图。

Gephi 能够计算网络的常用指标，包括中心性 (Degree,Betweenness,Closeness)、 密度 (Density) 、 路径长度 (Path Length)、半径 (Diameter) 、 集聚系数 (Clustering Co- efficient) 、模块度 (Modularity) 等。它提供了最短路径 (Shortest Path) 、PageRank、 HITS 等算法。Gephi 还支持社区检测 (Community Detection, 基于Modularity 的方法)。 最后，Gephi 能够随机地生成 (Random Generator) 各种实验用的社交网络。

**8.4.2 UCINET**

UCINET 是一款知名的、功能强大的社交网络综合性分析工具。这款软件由 Lin Freeman,Martin Everett,Steve Borgatti等几位作者共同开发。 UCINET 包含了Net- Draw 工具，用于网络的二维可视化。它还将集成 Mage 可视化工具，用于网络的三维可 视化。此外，UCINET 还可以集成Pajek, 用于大型社交网络的分析。

UCINET 包含了大量的网络分析例程，用于计算网络的指标，比如中心性、检测各种 类型的网络子结构 (Sub Structure) 和网络社区、进行自我网络分析 (Ego Network A- nalysis) 等。

UCINET 还支持聚类分析、多维尺度分析 (Multi dimensional Scaling)、奇异值分解 (Singular Value Decomposition)、因子分析 (Factor Analysis)、对应分析 (Correspon- dence Analysis)、核心-边缘结构分析、地位和角色分析等。其中，核心-边缘 (Core-Pe- riphery) 结构分析的目的是研究社交网络中，哪些节点处于核心地位，哪些节点处于边缘 地 位 。

**8.4.3 Pajek**

Pajek 是一款大型网络分析和可视化软件。网络规模可以达到10亿级别的顶点，边的 数量则没有限制(除非内存空间不够)。Pajek 自1996年开发出第一个版本以来，历经了 20多个年头。它是一款免费软件，用户可以自由下载和使用软件、文档以及其他材料。

Pajek 支持主流的网络布局算法，包括 Kamada-Kawai 优化算法、Fruchterman Rein- gold优化算法、VOS 映射 (Mapping) 算法、Pivot MDS、鱼眼变换 (Fish Eye) 以及按 照图层进行绘制 (Drawing in Layers) 等。

Pajek 支持丰富的社交网络分析功能，包括计算顶点的中心性 (VertexCentrality) 和 网络的中心势 (Centralization of Network)、顶点的 Hub 和 Authority、网络的集聚系数 (Clustering Coefficient)、检测网络社区 (Community Detection)、查找各种联通组件 (Connected Component,包 括 Weakly,Strongly,Bi-connected) 、 查找最短路径 (Shor- test Path)、查找 K 近邻 (K-Neighbor) 、 查找最大流 (Maximum Flow)、查找结构洞 (Structural Hole)、生成各种随机网络 (Random Network) 等众多的功能。

**8.4.4 NodeXL**

NodeXL(Network Overview,Discovery and Exploration for eXceL) 是微软提供的

一款开源的社交网络可视化工具软件。 NodeXL 遵循微软公开协议 (Microsoft Public Li- cense), 可以免费使用和分发。它提供高度的可编程性，用户可以在NodeXL 之上进行二 次开发，通过编程调用NodeXL 在 Excel 中提供的全部功能。NodeXL 有两个版本： 一个 是 NodeXL Basic; 一个是NodeXL Pro。

NodeXL Basic的主要目的是进行社交网络的可视化，NodeXL Pro提供了很多附加的 功能，极大地扩展了NodeXL Basic。这些功能包括：(1)存取社交网络数据流；(2)计 算网络指标 (Network Metric);(3) 文本和情感分析；(4)报表生成 (Report Genera- tion) 等。

NodeXL Basic和 ExcelR 2007 、2010 、2013或者2016等版本深度集成。和微软的其 他优秀产品一样，NodeXL 容易学习，使用非常方便。NodeXL 作为一个Excel 模板安装， 安装完成后，根据 NodeXL Graph 模板新建一个 xls 文档就可以使用了。用户需要在 “Vertex 工作表”中填人顶点的相关信息，在 “Edge 工作表”中填人边的相关信息，单击 “Show Graph” 按钮，系统按照自动布局方法把图显示出来。

NodeXL Basic支持多种输入格式，可以直接导人流行的UCINET,Pajek,GraphML 等文件格式，或者最简单的邻接矩阵等，它也可以导出到这些文件格式。

用户可以设定顶点的颜色、形状、大小、标签、透明度等属性，手工拖动顶点对图的 可视化效果 (Appearance) 进行修改，以及对显示结果进行缩放 (Zoom &Scale)。

用户可以选择显示整个网络或者对节点进行过滤 (Dynamic Filtering), 只显示符合条 件的节点(比如只显示度大于5的节点等),以及针对某个节点显示自我网络 (Ego Net- work)。

用户可以使用NodeXL 计算一些网络的基本指标 (Metric, 使用斯坦福大学的SNAP 库)。NodeXL Basic 仅支持简单的指标计算，比如节点的度 (Degree) 等 。NodeXL Pro 则可以计算 Betweenness 中心性、 Closeness 中心性、 Eigenvector中心性、集聚系数 (Clustering Coefficient) 、PageRank、图的密度 (Graph Density) 等。

NodeXL 可以根据某个属性对节点进行聚类，或者由NodeXL 自行分析顶点的连接关 系，自动把顶点聚类成类簇。各个类簇通过节点的颜色、形状进行区分。各个类簇还可以 用不同的框标示出来。类簇可以进行折叠 (Collapse), 而类簇间的连接可以归拢 (Bun- dle) 起来，以便观察。

除了上述软件，用于社交网络可视化以及分析的软件还有Cytoscape①,Visone②, SocNetV③,CiteSpace④ 等。

**8.5** **思考题**

(1)社交网络的应用、社交网络的类型、社交网络的表示方法。



①<http://www.cytoscape.org/.>

②<http://visone.info/html/demo.html.>

③<http://socnetv.sourceforge.net/.>

④<http://cluster.cis.drexel.edu/>～cchen/citespace/.



(2)小世界模型、 BA 模型、ER 模型。

(3)网络的基本属性。

(4)节点的中心性。

(5)PageRank 算法原理及其应用、HITS 算法原理及其应用。

(6)可达性、路径、最短路径、最小生成树、主路径分析。

(7)凝聚子群与社区检测。

(8)链路预测、信息扩散与影响力分析。

(9)核心-边缘分析。

**(10)位置和角色、子图查询、网络模体。**

I