自然语言处理

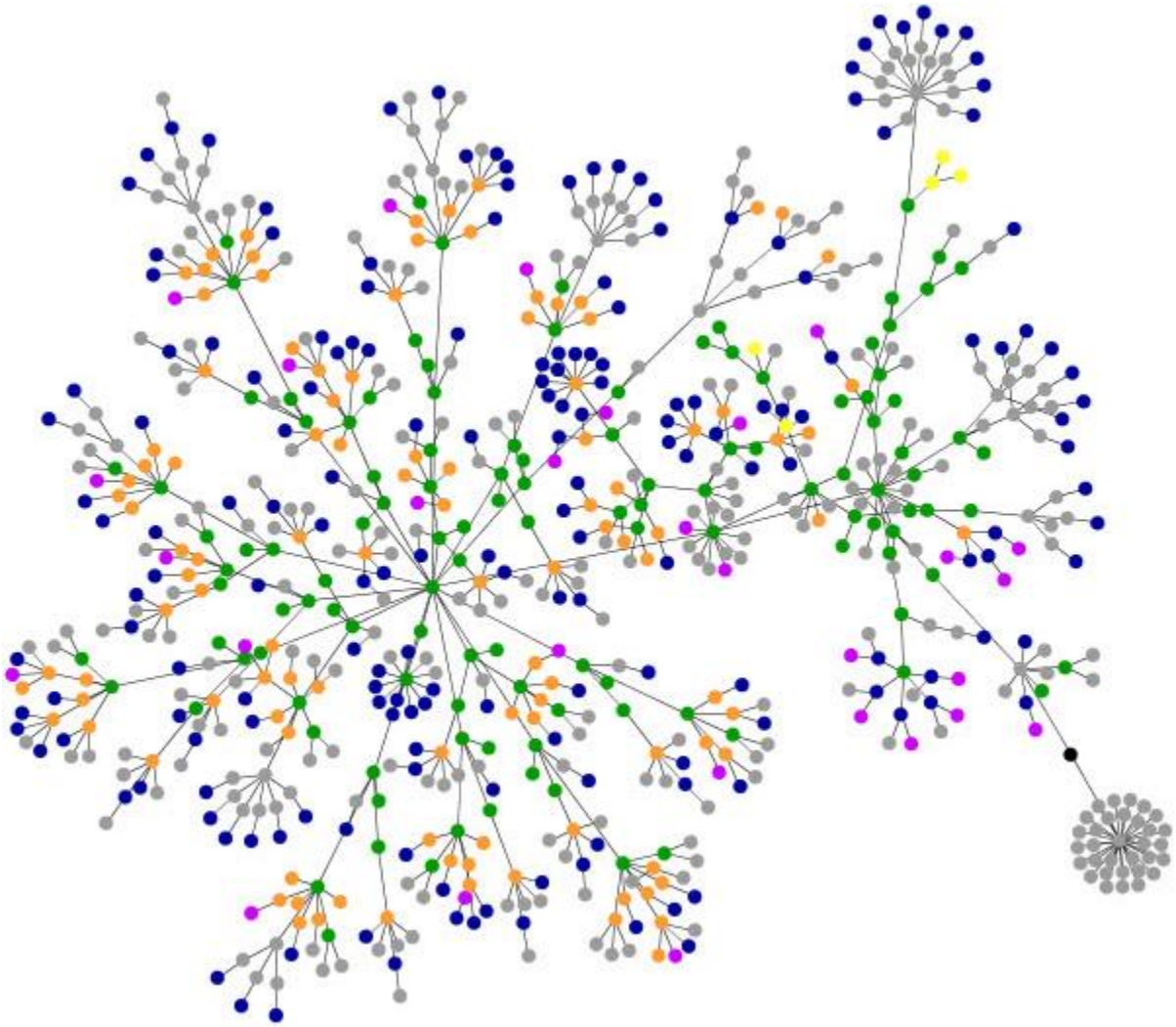
Link Analysis, PageRank

创新港

大纲

- 锚文本
- Web链接分析
- PageRank

Web图



Web图

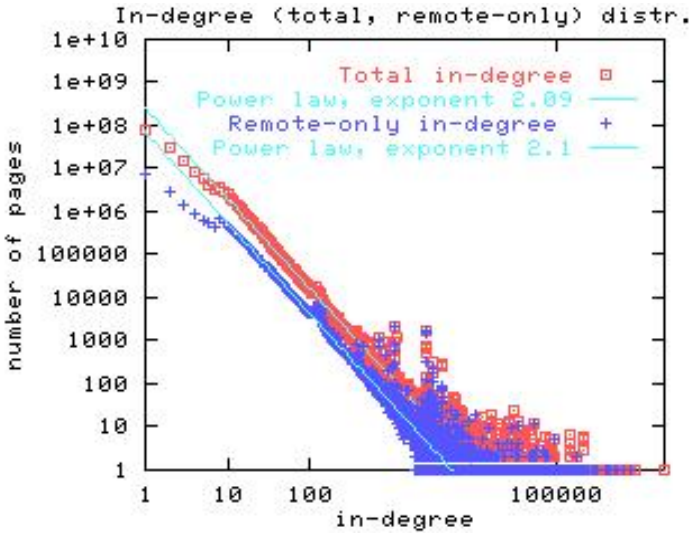
- 可以使用有向图表征Web网络结构
  - 节点：网页（或网站）
  - 有向边：入链接、出链接
- 特点：
  - 动态、持续变化
    - 增加节点、边
    - 删除节点、边
    - 动态生成的节点和边（on the fly）
  - 时序问题
    - 变化增长率
    - 生命期

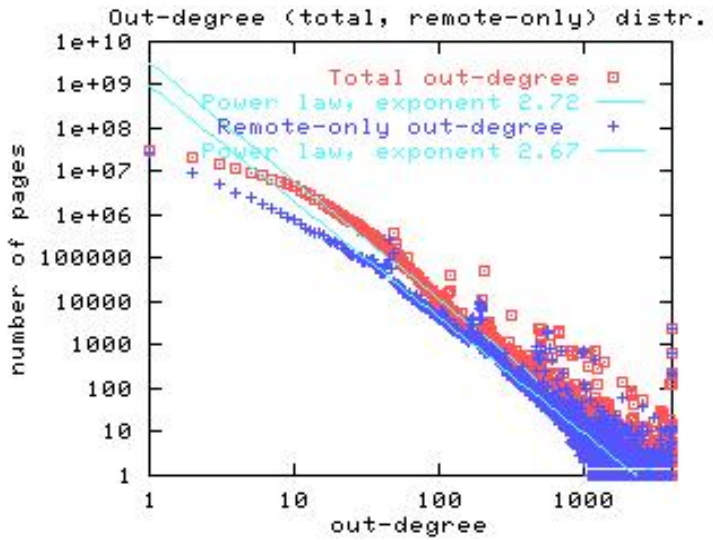
Web图

- 图的大小(Size)
  - 总的节点数目
    - 很难估计，一个下界：搜索引擎索引的网页数(crawled pages)
  - 总的边数目：
    - 更难估计，自动生成的链接大量存在
- 连通性(connectivity)
  - 强连通子图（SCC）：子图中任意两点u、v，存在有向路径u→v和v→u，可以互相连接
  - 弱连通子图（WCC）：子图中任意两点u、v，在忽略边方向时，存在无向路径可以互相连接

Web图

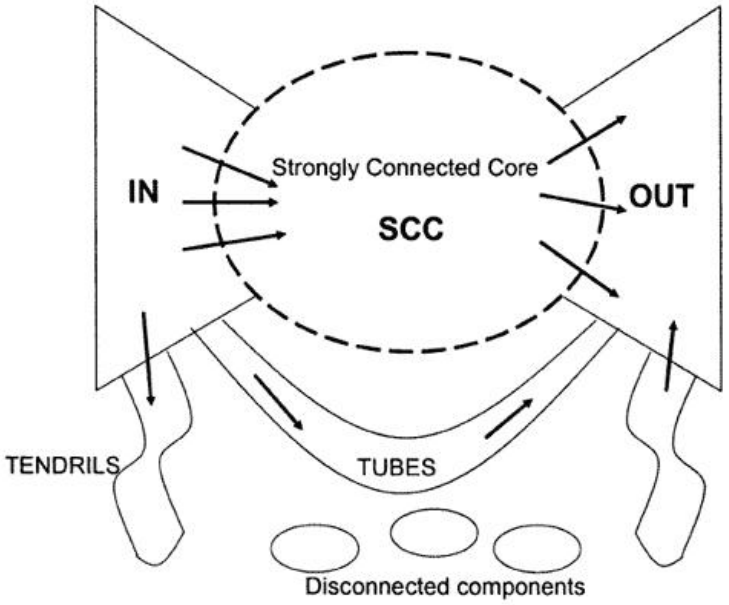
- Web图特性：
  - 幂率分布
    - 节点数和边数服从幂律分布（power law distribution）





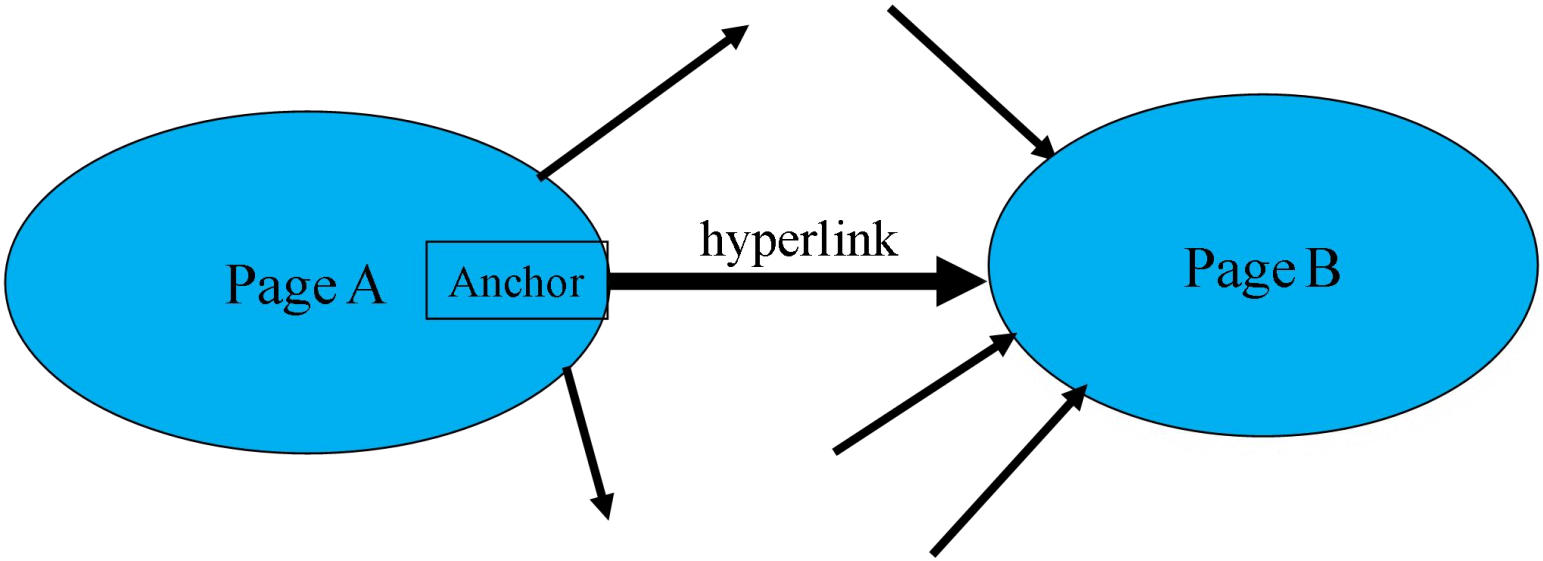
Web图

- Web图特性：
  - 领结结构（Bow-tie Components）



  - SCC部分的大小只占整个 Web 的一小部分
    - Web上大量充斥的是那些没多少“价值”的网页

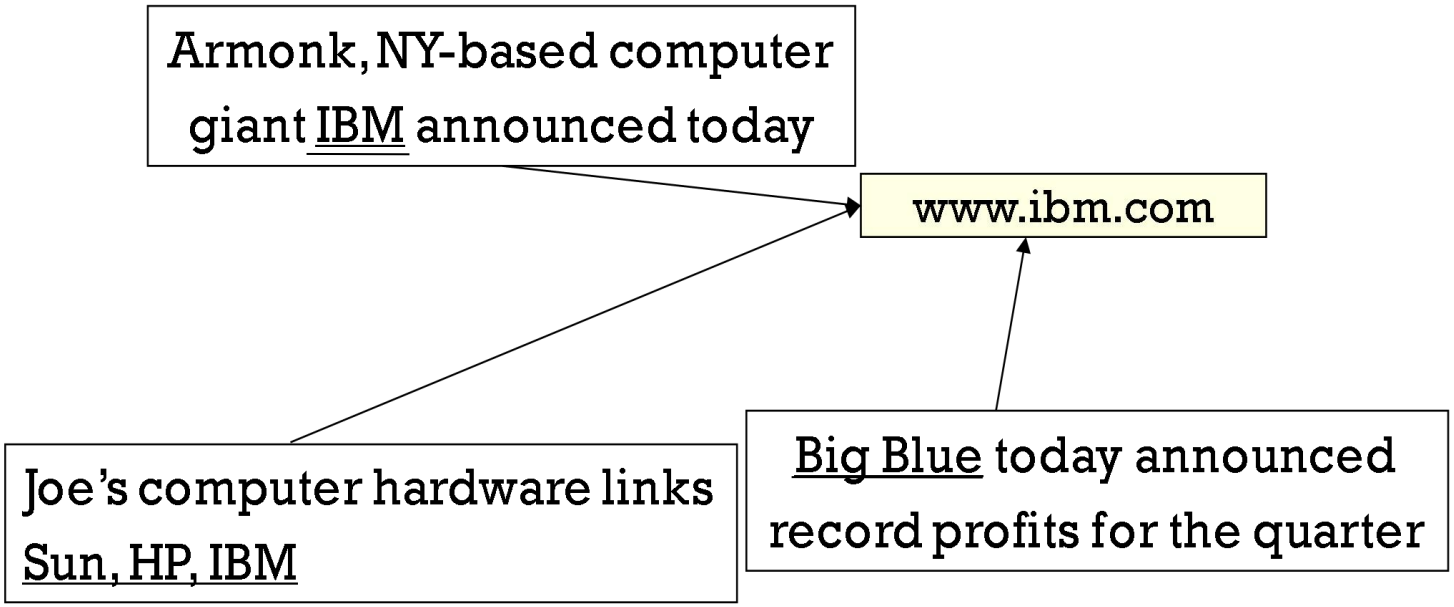
锚文本



- 锚文本（anchor）：即超链接上可以点击的文本
- 假设1：内容推荐假设
- 假设2：主题相关假设

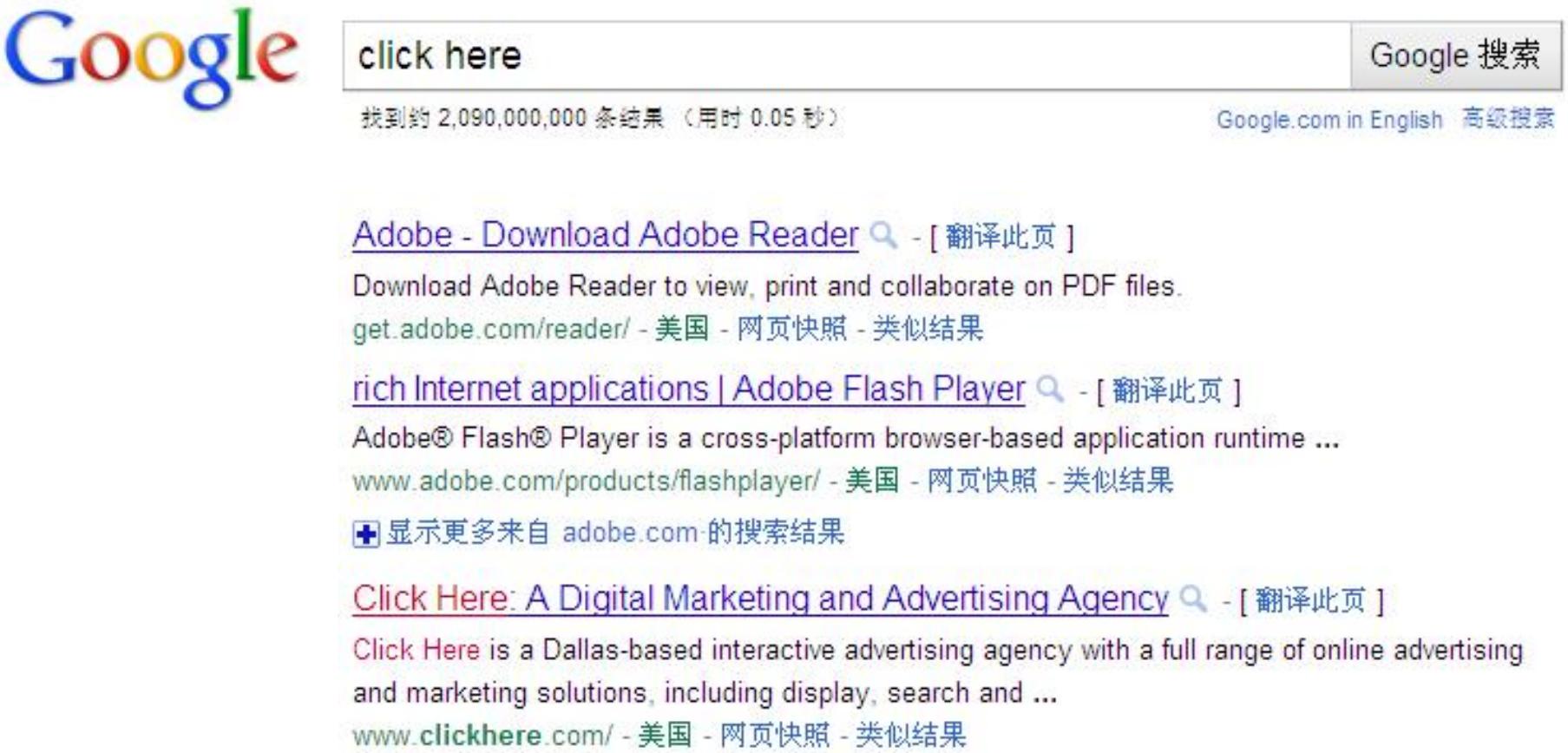
锚文本

- 索引锚文本：为文档D构建索引时，指向它的超链接上的锚文本也作为D的索引项



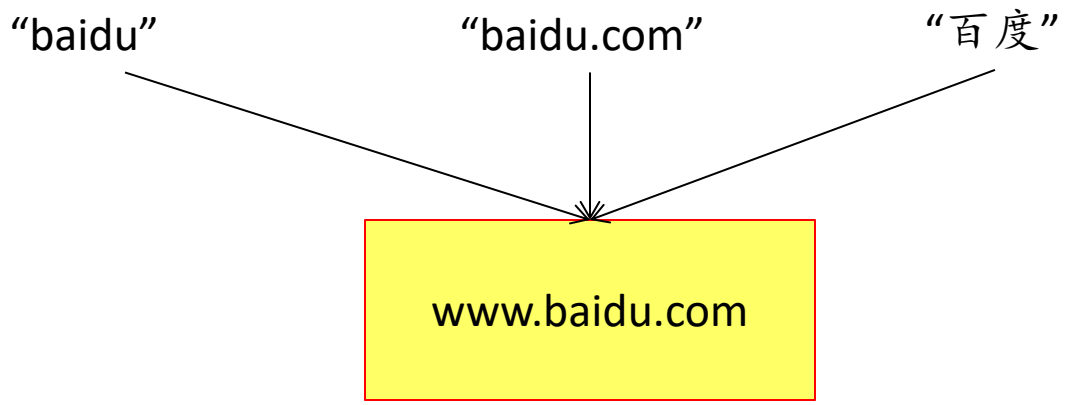
- 很多锚文本周围的文本常常也可以当成锚文本使用
  - “北京高考最新相关信息”
  - 思考：对于某个网页，假设给定了它所有锚文本集合，如何从这个集合中选出针对x的最具描述性的锚文本？

锚文本


Google
click here
Google 搜索
找到约 2,090,000,000 条结果 （用时 0.05 秒）
Google.com in English 高级搜索
Adobe - Download Adobe Reader - [翻译此页]
Download Adobe Reader to view, print and collaborate on PDF files.
get.adobe.com/reader/ - 美国 - 网页快照 - 类似结果
rich Internet applications | Adobe Flash Player - [翻译此页]
Adobe® Flash® Player is a cross-platform browser-based application runtime ...
www.adobe.com/products/flashplayer/ - 美国 - 网页快照 - 类似结果
显示更多来自 adobe.com 的搜索结果
Click Here: A Digital Marketing and Advertising Agency - [翻译此页]
Click Here is a Dallas-based interactive advertising agency with a full range of online advertising and marketing solutions, including display, search and ...
www.clickhere.com/ - 美国 - 网页快照 - 类似结果

锚文本

- 搜索词条“百度”，需要及如何区分：
  - Baidu的主页 (homepage)
  - Baidu版权的页面（copyright page，其中“百度”词频很高）
  - Baidu竞争对手的页面（也可能多次提及“百度”）



链接分析

- 锚文本对于其所指向网页的说明，是否具有相同的作用
  - 实际情况：一些网站上的内容，其可信度及权威较高，而一些网站上的内容，其可信性及影响力较差
  - 需要对网页的“权威性”、“影响力”进行评估
- 链接分析的目的：
  - 利用HTML网页的链接特点，改善查询的效果
  - 基于超链接结构对网页的质量/重要性进行评估

链接分析

- 输入：超链接结构图
- 输出：页面重要性/质量得分
- 对质量的定义不同：
  - PageRank、Inverse-PageRank、Browse-PageRank
  - HITS
  - TrustRank
  - …

PageRank的出发点

- 社交网络中的“声望模型”：
- 给定一个群体S，及其上的“知晓”关系R，便定义了一个有向“关系图”G
- 用邻接矩阵P表示，$P_{ij}=1$当且仅当i“听说过”j（注意这里没有程度之分）
- 我们希望确定p_i：所有个体$i \in S$的“声望”
- 模型一：$p_i = \sum P_{kj}$，$k=1,\ldots,n$，即i在G上的“入度”，亦即P的第i列的1的个数
  - 清楚、好计算；但是“不够好”
- 模型二：$p(i) = \sum P_{kj}p(k)$，$k=1,\ldots,n$，即i的声望等于知晓他的人的声望之和
  - 清楚、显得要更“精确些”；但是“如何计算”？

PageRank的出发点

- 给每个网页一个分值，以衡量它在网页集合中的相对重要程度
  - 将指向网页的超链接看成对网页的“投票”
  - 网页的PageRank值定义为一个递归的变量，取决于指向它的网页（入链接）的PageRank值

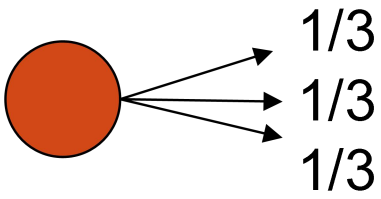
$$PR(u) = \sum_{v \in B_u} \frac{PR(v)}{L(v)}$$

Bu: 指向u的链接
L(*v*): 页面v的出链接个数

- 从链接分析的角度：
  - 把超链接关系作为一个“投票”动作，获得较多投票的网页质量较高
  - “不公平”的“民主决策”过程

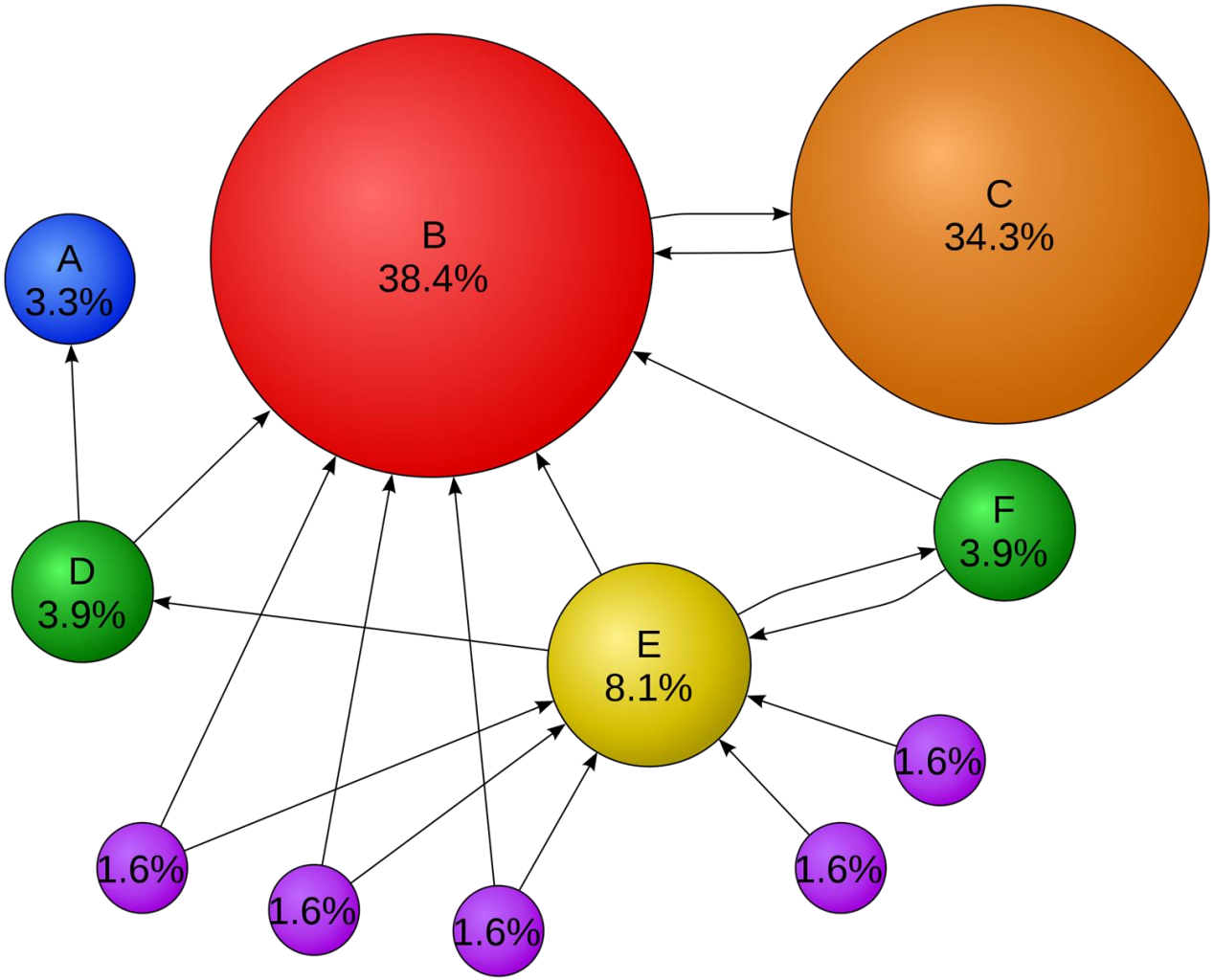
PageRank的出发点

- 随机行走模型：设想一个用户在浏览网页时，随机选择（equiprobably）一个链出的链接继续访问，永不休止……



- 则足够长的时间后（稳态情况下），他会正在看哪一篇网页？
- 稳态情况下：每个网页*u*都会有一个访问概率PR(*u*)，即PageRank值，作为网页重要程度的度量
- PR(*u*)依赖于上一个时刻到达“链向”*u*的网页的概率，以及这些网页中出链接的个数

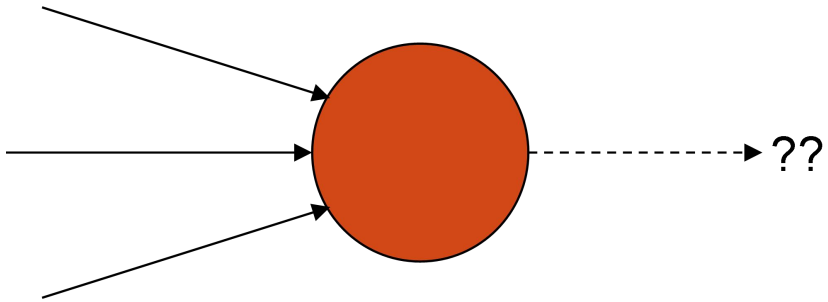
PageRank



From Wikipedia: 尽管E有更多的入链接，但C的PageRank值大于E

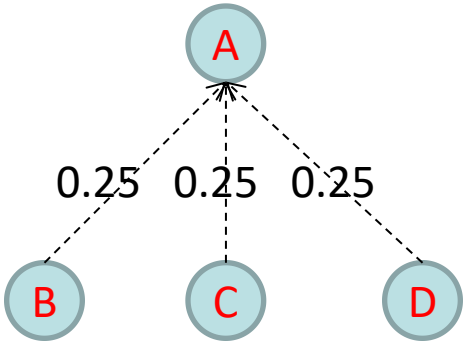
PageRank

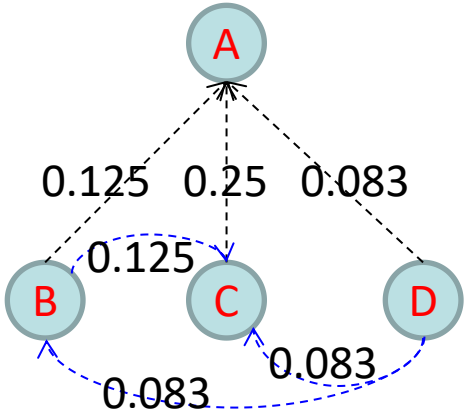
- 随机行走模型并不够好....
  - 上述“随机浏览”模型有效的条件是：由网页形成的有向图允许通过链接关系访问到每一个网页
  - 但有三种情况是破坏这条件的
    - 图中形成“圈”(rank bounces)
    - 有入度为零的点(rank isolates)
    - 有出度为零的点(rank sinks)



例子

- 初始化： PR(A)= PR(B)= PR(C)= PR(D)=1/4



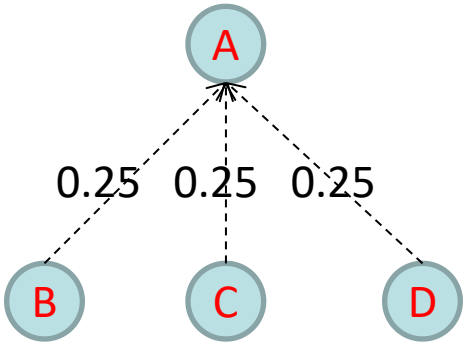


$$PR(A) = PR(B) + PR(C) + PR(D).$$

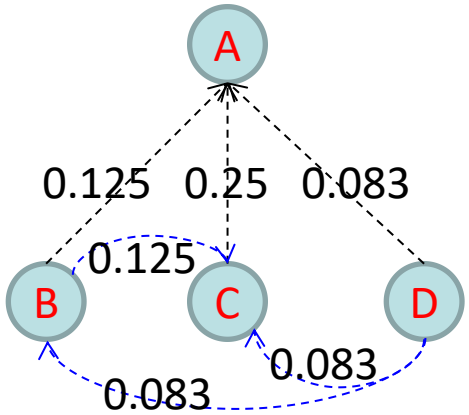
$$PR(A) = \frac{PR(B)}{2} + \frac{PR(C)}{1} + \frac{PR(D)}{3}$$

例子

- 几轮迭代之后：



	PR(A)	PR(B)	PR(C)	PR(D)
0	0.25	0.25	0.25	0.25
1	0.75	0	0	0
2	0	0	0	0



	PR(A)	PR(B)	PR(C)	PR(D)
0	0.25	0.25	0.25	0.25
1	0.4583	0.0833	0.2083	0
2	0.25	0	0.0417	0
3	0.0417	0	0	0
4	0	0	0	0

结束于 (0, 0, 0, 0)...

随机跳转（teleporting）

- 基本假设：用户在浏览页面时，从Web图中的一个节点随机地跳转到另一个节点
  - 若N为Web图中的节点个数，在用户选择页面时，跳转操作保证每个页面被访问的概率为 1/*N*
- 具体实施：
  - (1) 若一个节点出度为零（出链接个数为零），则用户激活跳转操作
  - (2)若一个节点出度不为零，则用户以概率α（$0 < \alpha < 1$）激活跳转操作，而以概率$1-\alpha$进行随机行走，即等概率地选择一个出链接指向的页面
    - 其中α为预先设置的参数，比如$\alpha=0.1$

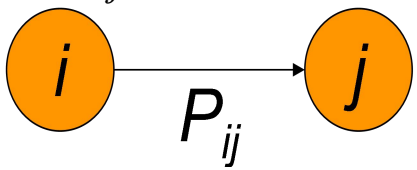
随机跳转（teleporting）

- 跳转操作的作用：
  - 跳出局部受困状态
  - 长远来看，每个页面都有被访问的几率
  - 如何计算这个被访问的几率？
- 当用户执行混合过程时（随机行走加上随机跳转），他就会以一个固定的时间比例$\pi(v)$访问每个节点v，$\pi(v)$取决于：
  - (1) Web图结构
  - (2) α的值
- 采用马尔科夫链理论来说明

马尔科夫链

- 一个马尔科夫链（Markov chains）包含：
  - N个状态
  - N*N的转移概率矩阵P
- 在任一步，马尔科夫链都处于N个状态中的一个
- 对于1≤ i, j ≤ N，概率P_{ij}表示从当前状态i转移到下一个状态j的条件转移概率

$$\sum_{j=1}^{N} P_{ij} = 1$$



马尔科夫链

- Web图的邻接矩阵*A*可定义为：如果页面i和j之间存在超链接，则 $A_{ij} = 1$，*否则*$A_{ij} = 0$
- 从矩阵A可以推导出马尔科夫链的*转移概率矩阵*P：
  - 若A的某一行没有1（即该行对应的网页出链接个数为0），则用 1/N代替每个元素；
  - 对于其它行，处理如下：
    - (1) 将1除以该行中1的个数（如果某行有3个1，则每个1用1/3代替）
    - (2)将(1)中得到的矩阵乘以系数$1 - \alpha$
    - (3)将(2)中得到的矩阵每个元素都加上 α/N
  - 得到矩阵P

马尔科夫链

- 概率向量：用一个行向量 $x= (x_1, \ldots x_n)$ 表明用户当前在浏览的页面
  - E.g., (000…1…000) 表示当前在页面i
- 更一般地，向量 $x= (x_1, \ldots x_n)$ 表明用户在浏览页面i的概率为 x_i

$$\sum_{i=1}^{n} x_i = 1$$

马尔科夫链

- 设用户在任意时间所访问页面的概率分布用向量x来描述
- t=0时，用户处于某个初始状态，x中与其对应的元素为1，其它为0
- t=1时，用户访问页面的概率分布为xP
- t=2时，用户访问页面的概率分布为$(xP)P=xP^2$
- 依此类推下去，可以计算用户在任意时刻访问页面的概率分布
- 两个问题：
  - 是否会收敛？
  - 收敛条件是什么？

马尔科夫链

- 遍历马尔科夫链（Ergodic Markov chain）
  - 如果存在着一个正整数T_0，使得对所有的状态i，j都满足：从初始状态i出发，在任意时刻$T>T_0$，处于状态j的概率不为零
  - 即：从任意两个状态之间都存在一个通路（虽然可能不止通过一步转移）

马尔科夫链

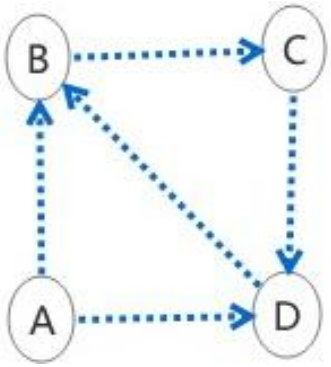
- **定理：**对任意遍历马尔科夫链，都存在一个唯一的稳态概率向量π，它是矩阵P的主（左）特征向量，并且，如果η(i, t)是在t步之内状态i的访问次数，那么有：

$$\lim_{t \to \infty} \frac{\eta(i,t)}{t} = \pi(i)$$

  - 其中，π(i) > 0 是状态i的稳态概率

- 一个长时间段后，访问页面i的可能性正比于这个比值π(i)
- π(i) 与起始状态无关

稳态概率向量



初始化 PR 向量 x = $\left(\frac{1}{4} \quad \frac{1}{4} \quad \frac{1}{4} \quad \frac{1}{4}\right)$

转移概率矩阵 P = $\begin{pmatrix} 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}$

	PR(A)	PR(B)	PR(C)	PR(D)
0	0.25	0.25	0.25	0.25
1	0	0.375	0.25	0.375
2	0	0.375	0.375	0.25
3	0	0.25	0.375	0.375
4	0	0.375	0.25	0.375
5	0	…	…	…

随着迭代的进行，PR向量趋于一个稳定值！

稳态概率向量

- 用x表示当前位置的概率向量，则下一位置的概率向量为 xP
- 用概率向量 a= (a_1, … a_n)表示稳态概率分布
  - a_i 表示处于状态i的概率
  - 例如，a_1=1/4 and a_2=3/4
- 若a为稳态概率向量，则会有aP=a
- 满足aP=a的a即为稳态概率向量
  - a是矩阵P的（左）特征向量
    - 转移概率矩阵P的最大特征值为1
    - a对应于P的最大特征值对应的主特征向量
    - 如何计算？

稳态概率向量

- 无论初始状态如何，最终会到达稳态a
- 计算过程：
  - 从某一分布出发（例如x=(1 0 … 0)）
  - 经过一次转移，到达 xP
  - 经过两次转移，到达xP^2，然后是xP^3 ….
  - 若k足够大，最终会有 $xP^k = a$
- 算法：向量x不断地右乘矩阵P，直至乘积趋于稳定
  - 对x不停的旋转缩放，直至得到一个稳定的向量

例子

- 设$\alpha = 0.5$
- 加入随机跳转后的转移概率矩阵为：

$$P = \begin{pmatrix} 1/6 & 2/3 & 1/6 \\ 5/12 & 1/6 & 5/12 \\ 1/6 & 2/3 & 1/6 \end{pmatrix}$$

$\vec{x_0} = (1\ 0\ 0)$

$\vec{x_0}P = \begin{pmatrix} 1/6 & 2/3 & 1/6 \end{pmatrix} = \vec{x_1}$

$\vec{x_0}$	1	0	0
$\vec{x_1}$	1/6	2/3	1/6
$\vec{x_2}$	1/3	1/3	1/3
$\vec{x_3}$	1/4	1/2	1/4
$\vec{x_4}$	7/24	5/12	7/24
...	...	...	...
$\vec{x}$	5/18	4/9	5/18

PageRank总结

- PageRank计算：
  - 依据Web中的超链接，构建矩阵P
  - 依据P计算稳态概率向量a
  - 每个元素a_i是个0-1之间的值，对应于页面i的PageRank值
- 实际应用中：
  - 基本的PageRank对网页的排序不考虑Query，因此实际中PageRank只作为Rank模型一个影响因子
  - 实际中计算PageRank时，采用更多其它的复杂特征
  - 普遍使用基于机器学习的Rank技术

PageRank总结

- 随机行走模型的实际拟合程度如何？
  - 我们往往会点击网页的“回退”按钮...
  - 用户的浏览行为总趋于一个较短的路径
  - 搜索引擎、书签、导航等使得用户的浏览行为并不是随机的
- PageRank的扩展
  - 面向主题的PageRank
  - Browse-PageRank
  - ……

Hyperlink-Induced Topic Search (HITS)

- HITS和PageRank的两个基本区别：
  - HITS针对具体查询、应用在查询时，PageRank独立于查询，可应用在查询前
  - HITS算法考虑了不同链接的重要性，PageRank算法中对于向外链接的权值贡献是平均的
- R(q)：与查询q相关的网页集合
- V(q)：包括R(q)以及指向R(q)元素和被R(q)元素指向的网页
- 两个概念
  - AUTHORITY（权威型网页）：内容权威，质量高的网页
  - HUB（目录型网页）：指向许多authority网页的网页

HITS

- 声望高的（入度大）➔ 权威性高
- 认识许多声望高的（出度大）➔ 目录性强
- 针对$u \in V(q)$，在每个网页u上定义有两个参数：$a[u]$和$h[u]$，分别表示其权威性和目录性
- 交叉定义
  - 一个网页u的a值依赖于指向它的网页v的h值
  - 一个网页u的h值依赖于它所指的网页v的a值

HITS

- 计算过程：
  - 提交查询query
  - 选取检索结果集合R，将R，R所指向的网页和指向R的网页作为算法施行的网页集合G
  - 对G中每一个节点n，设定其内容权威度a(n)和链接权威度h(n)初始值均为1
  - 迭代计算：

$$a^{(k)}(n) = \sum_{m_i \to n} h^{(k-1)}(m_i)$$

$$h^{(k)}(n) = \sum_{n \to m_i} a^{(k)}(m_i)$$

PageRank & HITS

- 网页的相互链接特性，使得我们可以应用社会网络分析的方法来从网页集合的结构中提炼有用的信息
- 提炼什么信息？取决于我们对应用目标的认识，还取决于有关技术模型的精确定义、有效计算，以及对可能产生误差的认识和评估
- *PageRank*和*HITS*是两个经典的例子，大目标一致，切入点不同
- *PageRank*和*HITS*的一个本质缺陷是：将网页之间的链接关系“太当真”。

几个开源的搜索引擎框架

- Lucene：http://lucene.apache.org/
  - 开源的全文检索引擎工具包，是一个全文检索引擎的架构，提供了完整的查询引擎和索引引擎，部分文本分析引擎
- Nutch：http://nutch.apache.org/
  - Java实现的开源搜索引擎，包括全文搜索、Web爬虫等搜索引擎工具
- ……