5月21日12:00前，提交文献阅读相关素材
6月3日12:00前，提交实验报告及相关素材

信息检索与数据挖掘

第12章 Web搜索

4月29日，补充：概率图及主题模型
5月6日，补充：数据挖掘经典算法概述(1)
5月8日，补充：数据挖掘经典算法概述(2)
5月13日，第12章 Web搜索
5月15日，第13章 多媒体信息检索
5月20日，复习
5月22日，同学们文献阅读报告
5月27日，同学们文献阅读报告
6月3日，期末考试【暂定】

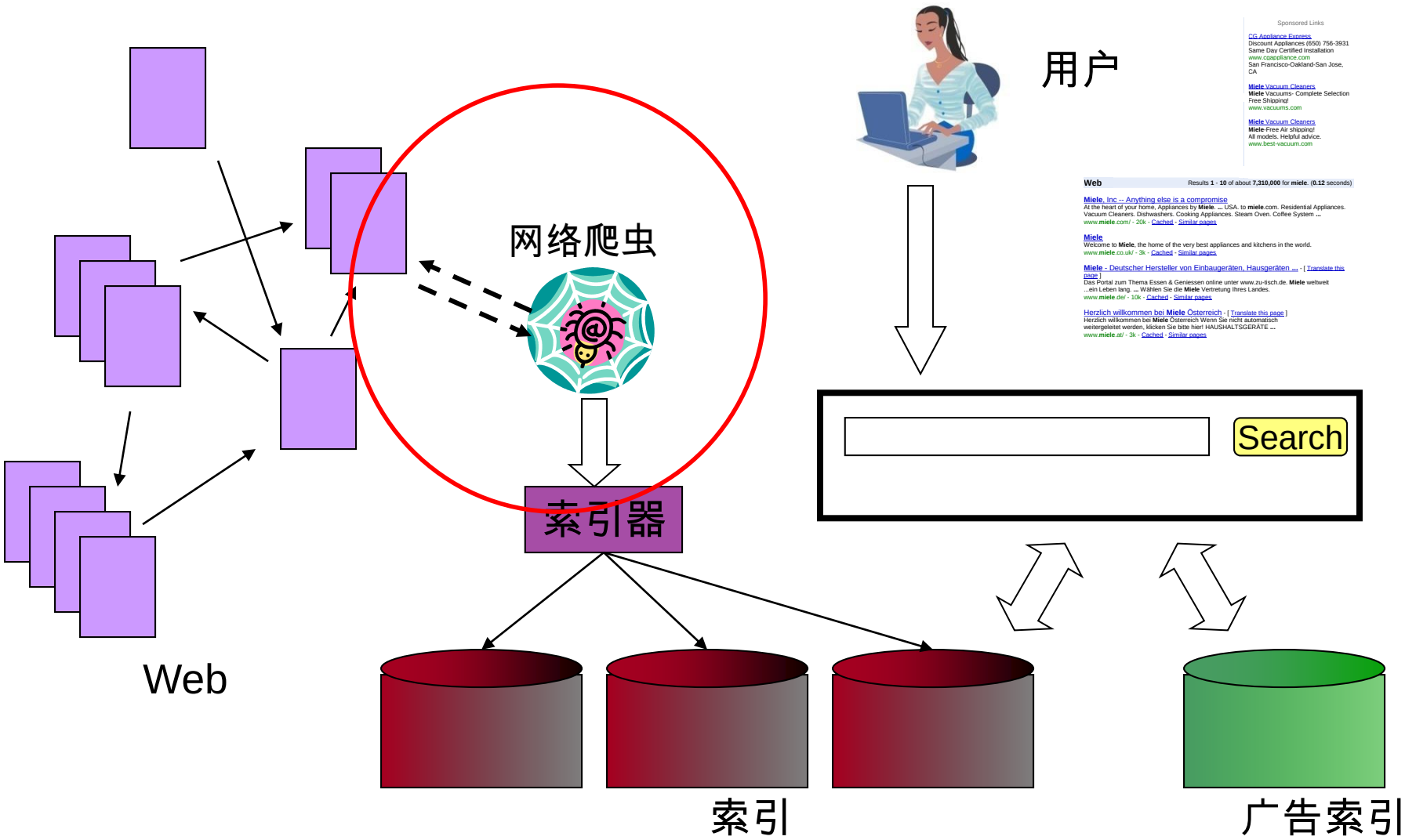
课程内容

- 第1章 绪论
- 第2章 布尔检索及倒排索引
- 第3章 词项词典和倒排记录表
- 第4章 索引构建和索引压缩
- 第5章 向量模型及检索系统
- 第6章 检索的评价
- 第7章 相关反馈和查询扩展
- 第8章 概率模型
- 第9章 基于语言建模的检索模型
- 第10章 文本分类
- 第11章 文本聚类
- 补充：概率图及主题模型
- 补充：数据挖掘经典算法概述
- 第12章 Web搜索
- 第13章 多媒体信息检索
- ~~第14章 其他应用简介~~

本讲内容：Web搜索

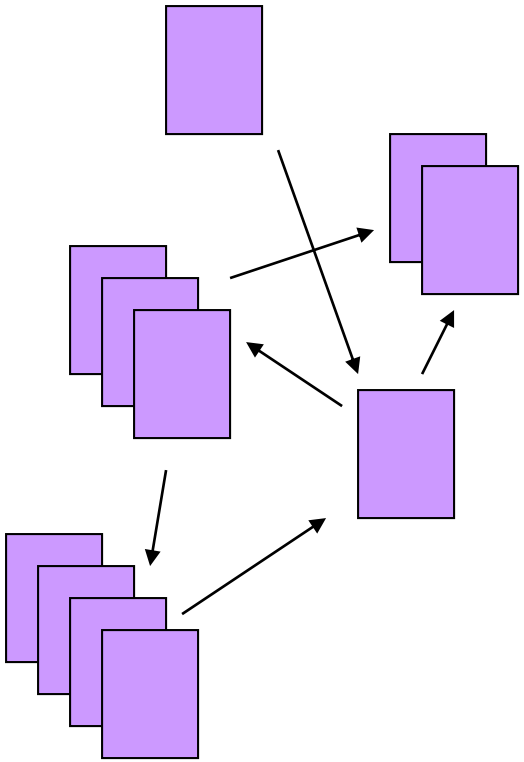
- Web采集
  - 采集器
  - 连接服务器
- 链接分析
  - 锚文本
  - 链接分析：Pagerank
  - 链接分析：HITS

Web搜索基本流程




http://zhidao.baidu.com/search
百度知道搜索_信息检索
转换
选择
fama2014sina
LV1
我的知道
消息
Baidu 知道
新闻 网页 贴吧 知道 音乐 图片 视频 地图 百科 文库 经验
信息检索
搜索答案
我要提问
共900,632条结果
筛选答案
信息检索_百度百科
信息检索（Information Retrieval）是指信息按一定的方式组织起来，并根据信息用户的需要找出有关的信息的过程和技术。狭义的信息检索就是信息检索过程的后半部分，即从信息集合中找出所需要的信息的过...
起源 | 定义 | 类型 | 主要环节
信息检索与利用试题题目
问：信息检索与利用试题题目 1.详细论述如何利用数据库查找《北京大学学报》...
答：信息检索答案 题型一 1、信息素养或素质的具体内容有那些？ 信息素质是指用户在利用以计算机及其网络技术为代表的现代科学技术进行知识学习、成长的过程中，逐步形成的主动参与信息活动、自觉应用信息技术的意识、态度、理念及具备的获娶识别、...
2014-07-03 3个回答
信息检索
答：第一章 信息：信息是事物存在的方式，运动状态及其特征的反映，是事物发出的信号，消息 信息的特征：载体依附性 无线共享性 永不枯竭性 开发增值性 应用时效性 存在普遍性 知识：知识是信息的升华和结果，系统化理论化的的信息就称为就称为知识 ...
2014-10-29 回答者: 望月曦澈 1个回答 3
什么是信息检索？
问：1.信息源的类型：个人信息源，组织机构信息源，实物型信息源，文献型信...
答：你想问什么？信息检索就像你搜索东西，得给出最合适的词条
2014-10-29 回答者: 454182560 3个回答
付费广告
算法生成的结果
推广链接
sci论文翻译机构哪家好?
上海熠逊生物技术有限公司主要提供SCI学术论文发表支持.基金申请咨询,标书修改,科研..
www.yxlwsci.com V1
什么是搜索引擎优化,首选百度!
什么是搜索引擎优化?选百度,覆盖95%网民,成为近50万家中国企业开展营销推广的首选!..
www.baidu.com V3
征信报告-Ease Credit
Easecredit 国内领先的在线B2B大数据服务公司.我们致力于提供更快捷和更低成本的在..
www.easecredit.cn V1
一心医译 十年成就 买医学文..
医药学大词典,医学文献王,用药参考为您的学习工作添动力.现在购买很优惠!
99yide.com V1
信息检索》来当当网,正..
买《信息检索》来当当网,正版低价,49元免运费,15年品质保证,购书首选!《信息检索》..
www.dangdang.com V3
搜索引擎用桌面百度,一键启..
百度公司全新推出桌面百度,更智能更快捷的

Web文档集

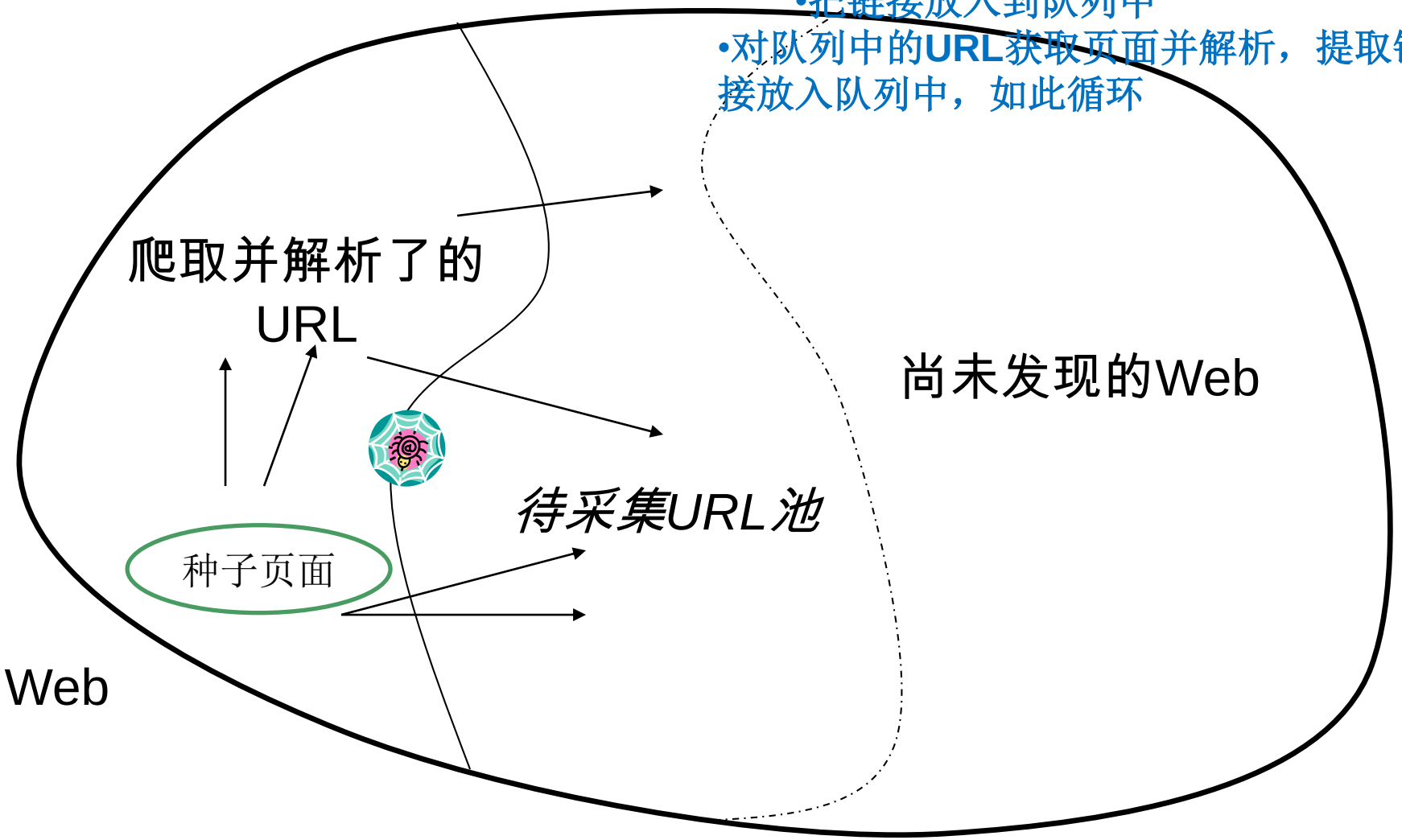


The Web

- 没有设计/多人协作
- 分散的内容创作、链接，民主化的发布
- 内容包含真理、谎言、矛盾和大量猜测 …
- 非结构化的(text，html，…)，半结构化的（XML，有注释的照片)，结构化的(数据库)…
- 规模比之前的文本集大得多… 但是其中有很多重复的记录
- 增长 - 最开始每几个月就翻一倍，现在涨速下降但依然在扩大
- *内容可能是动态生成的*

爬取过程

- 从已知的种子URL开始
- 获取页面并进行解析
  - 提取页面中包含的链接
  - 把链接放入到队列中
- 对队列中的URL获取页面并解析，提取链接放入队列中，如此循环



采集器必须具有的功能

- 礼貌性: Web服务器有显示或隐式的策略控制采集器的访问
  - 只爬允许爬的内容、尊重 **robots.txt**
- 鲁棒性: 能从采集器陷阱中跳出，能处理Web服务器的其他恶意行为
- 分布式: 应该可以在多台机器上分布式运行
- 可扩展性: 添加更多机器后采集率应该提高
- 性能和效率: 充分利用不同的系统资源，包括处理器、存储器和网络带宽
- 优先抓取“有用的网页”
- 新鲜度: 对原来抓取的网页进行更新
- 功能可扩展性：支持多方面的功能扩展，例如处理新的数据格式、新的抓取协议等

礼貌性

Robots.txt 源于1994年的协议，对爬取过程进行限制
http://www.robotstxt.org/orig.html 关于Robots.txt的说明

- 显式的礼貌: 根据网站站长的说明，选择允许爬取的部分进行爬取
  - 按robots.txt说的做，如下面写法的意思是：任何robot都不能访问 “/yoursite/temp/”开头的网址, 除了名叫“searchengine”的:

User-agent: *

Disallow: /yoursite/temp/

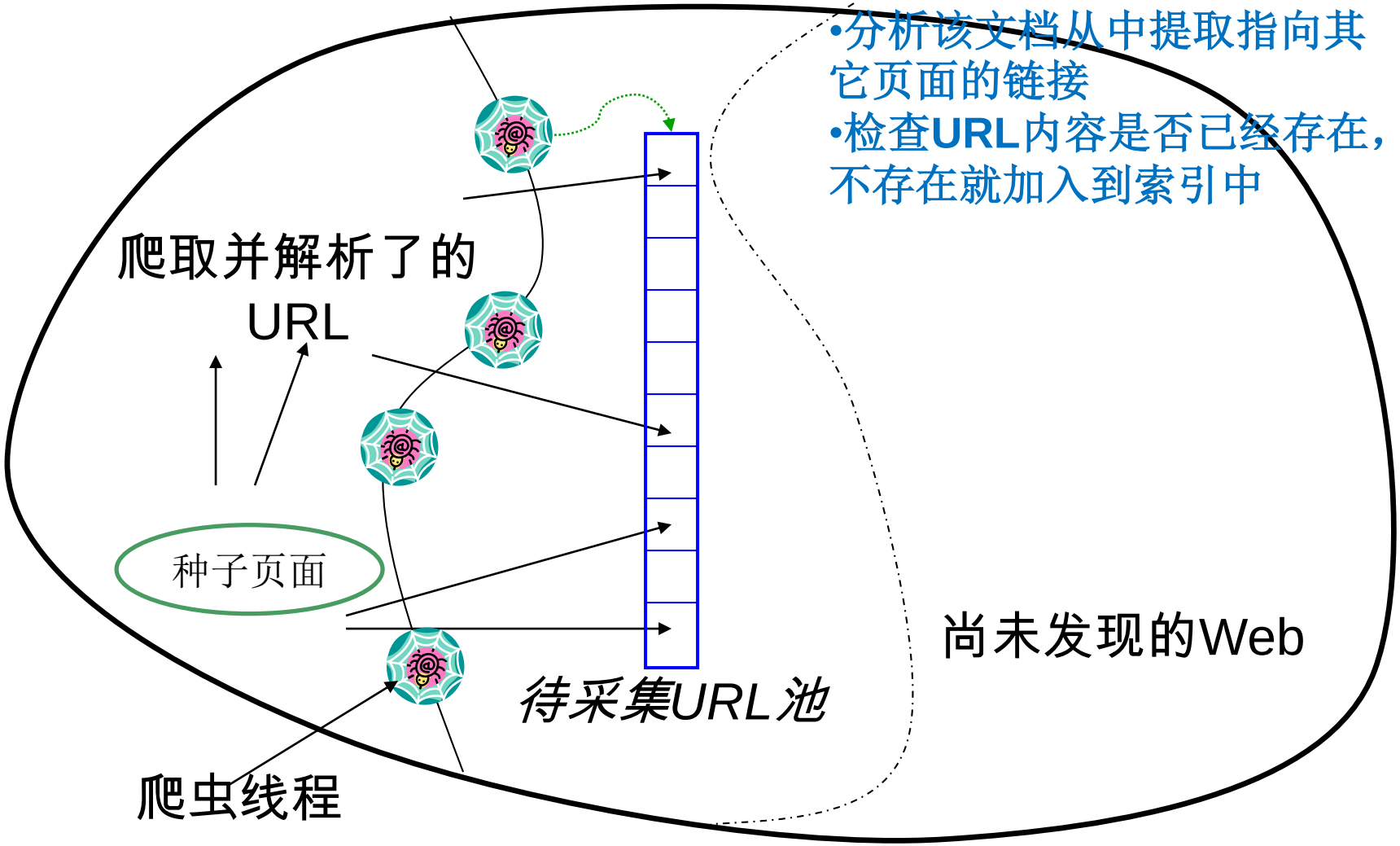
User-agent: searchengine

Disallow:

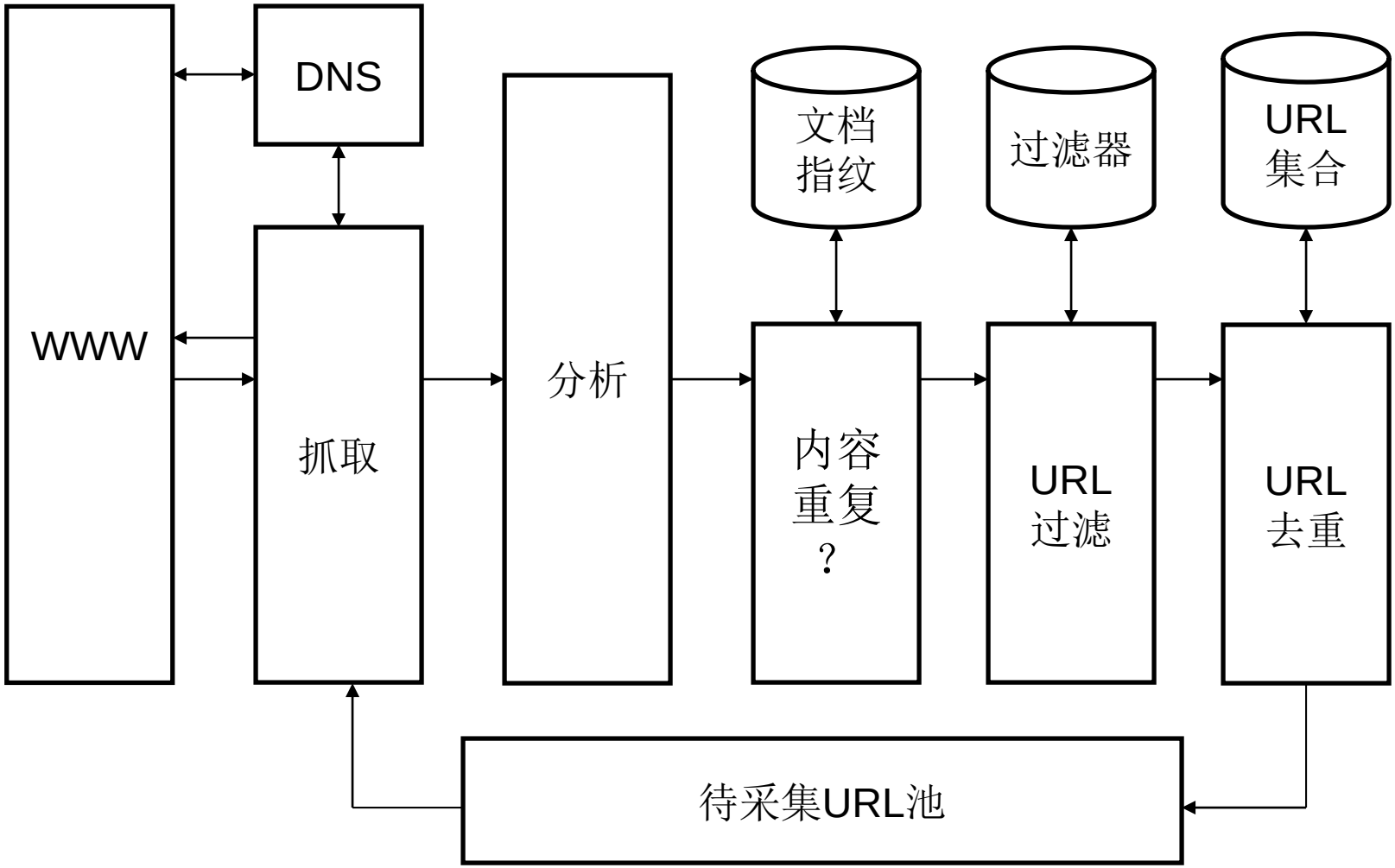
- 隐式的礼貌: 即使没有特别的说明，也不应该频繁的访问同一个网站

改进后的采集器

- 采集的过程
  - 多个
  - 从URL池中取一个URL：抓取URL对应的文档
  - 分析该文档从中提取指向其它页面的链接
  - 检查URL内容是否已经存在，不存在就加入到索引中



采集器基本架构

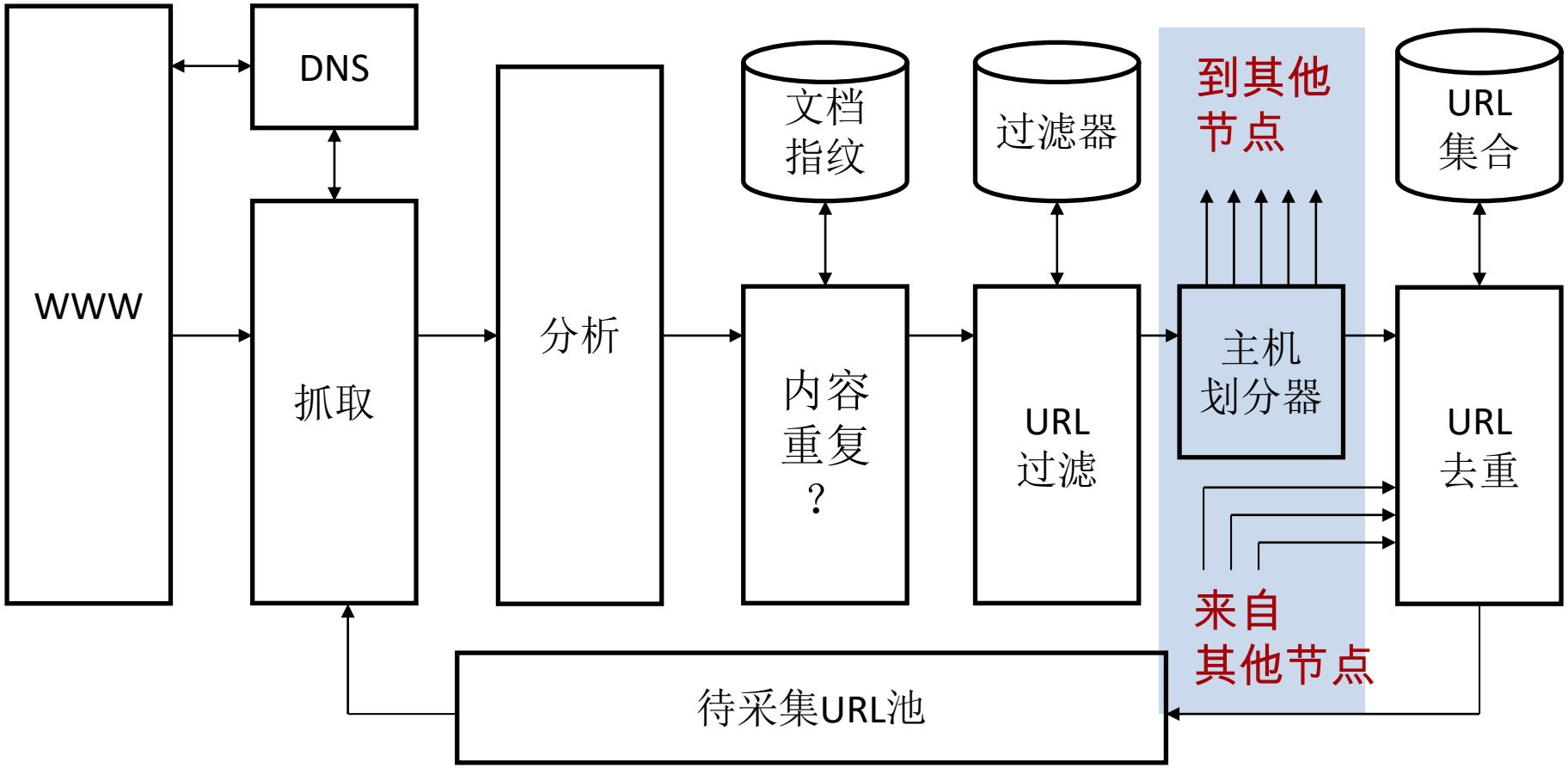


采集器分布化

- 在分布式系统环境下不同节点的不同进程中运行多个采集线程
  - 地理位置分布的采集系统
- 把要采集的主机分配到每个节点
  - 通过Hash函数或其他针对性的策略
- 不同节点之间怎么通讯？

节点间通信

- 通过过滤检测的URL需要发送到每个节点上进行查重处理



小结：采集器

- 礼貌性: Web服务器有显示或隐式的策略控制采集器的访问
  - 只爬允许爬的内容、尊重 **robots.txt**

```
User-agent: *
Disallow: /yoursite/temp/
User-agent: searchengine
Disallow:
```

- 鲁棒性: 能从采集器陷阱中跳出，能处理Web服务器的其他恶意行为
- 分布式: 应该可以在多台机器上分布式运行
- ……

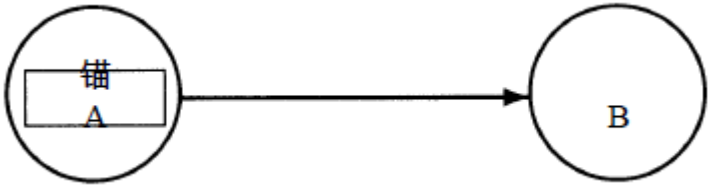
本讲内容：Web搜索

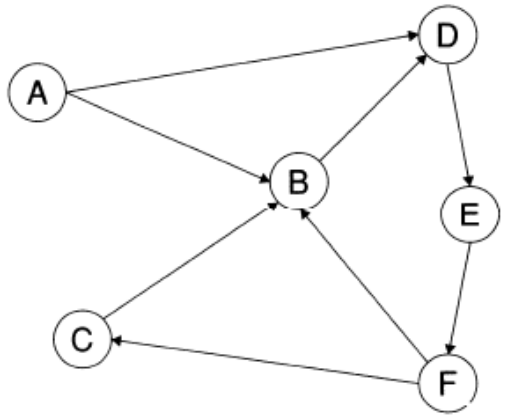
- Web采集
  - 采集器
  - 连接服务器
- 链接分析
  - 锚文本
  - 链接分析：Pagerank
  - 链接分析：HITS

Web 搜索引擎需要一个连接服务器（connectivity server）来支持Web 图连接查询（connectivity query）的快速处理。典型的连接查询包括“ 给定的URL 被哪些URL 所指向？”及“ 给定URL 指向了哪些URL？” 等。为此，我们在内存中存储了从URL 到出链及URL 到入链的映射表。

Web→Web图

- 我们可以将整个静态Web 看成是静态HTML 网页通过超链接互相连接而成的有向图，其中每个网页是图的顶点，而每个超链接则代表一个有向边。

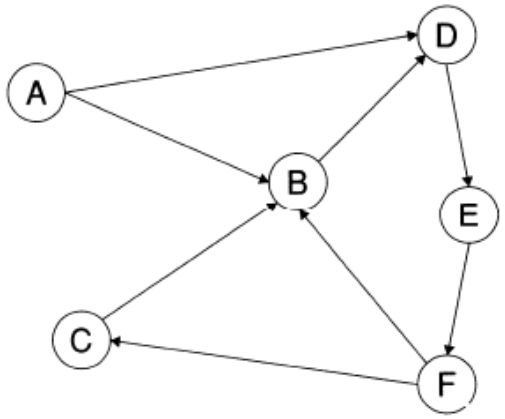




- 包含两个顶点A、B 的Web 图，每个顶点代表一个网页，A 网页上有一个超链接指向 B 。我们将所有这样的顶点和有向边集合称为Web 图。

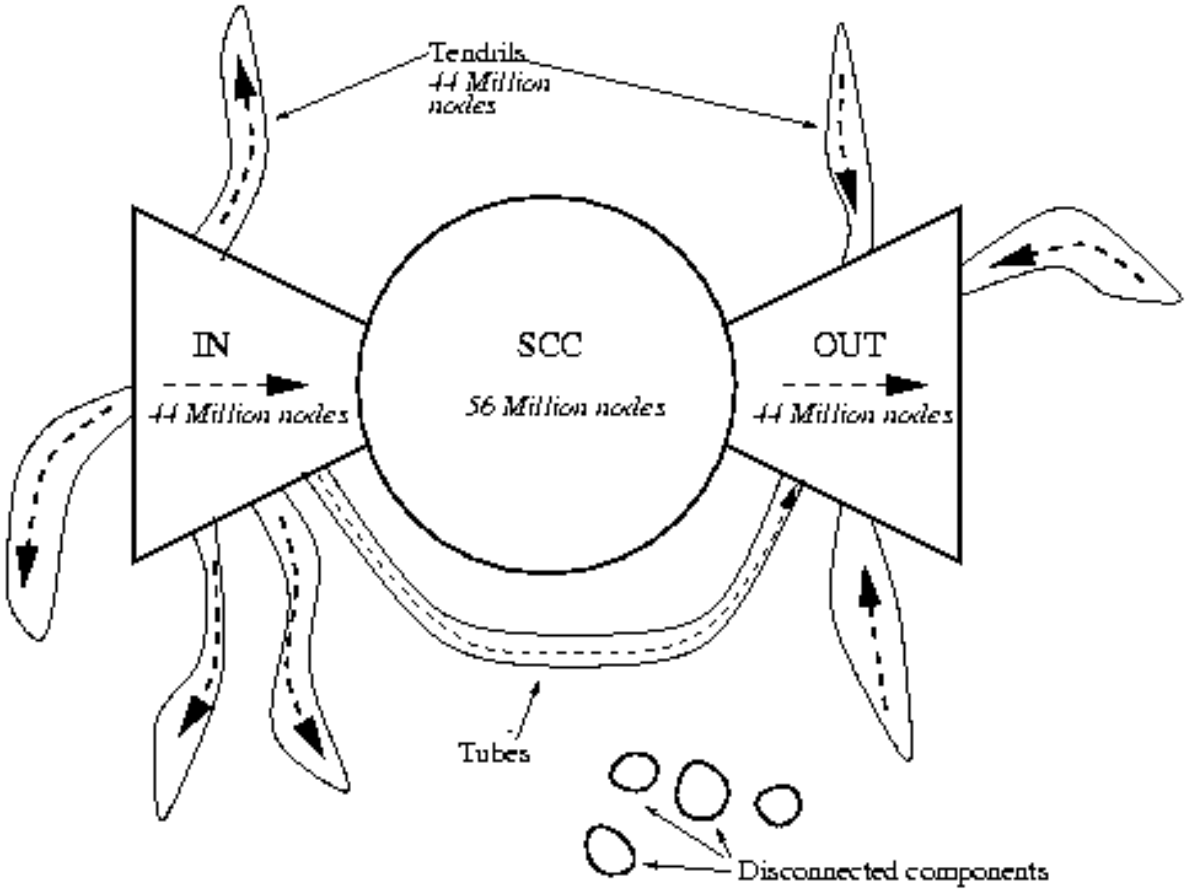
Web特性→ Web图特性

- 该有向图可能不是一个强连通（strongly connected）图，也就说，从一个网页出发，沿着超链接前进，有可能永远不会到达另外某个网页。我们将指向某个网页的链接称为入链接（in-link），而从某个网页指出去的链接称为出链接（out-link）。一个网页的入链接数目被称为这个网页的入度（in-degree），在一系列研究中得到的网页的平均入度大概从8 到15 左右不等。同样，我们可以定义某个网页的出链接数目为其出度（out-degree）。



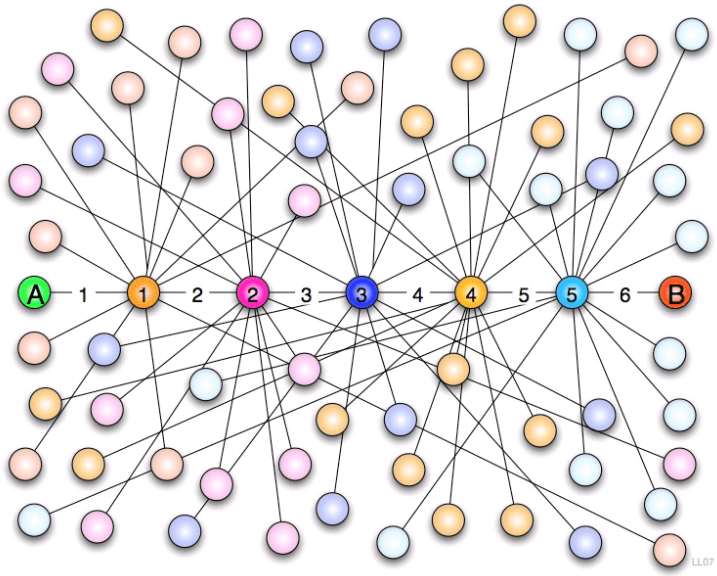
6 个网页（分别以A 到F 标识），网页B 的入度为3、出度为1。该图不是强连通图，因为B 不可能到A

Web特性→ Web图特性
Web的形状



- **Strongly Connected Component (SCC)**
  - Core
- **Upstream (IN)**
  - Core can't reach IN
- **Downstream (OUT)**
  - OUT can't reach core
- **Disconnected**
- **Tendrils & Tubes**

Web特性→ Web图特性
小世界网络



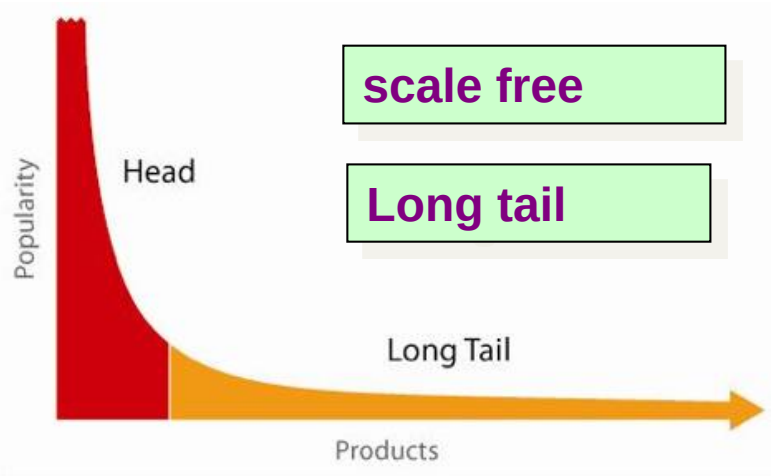
It's a small world

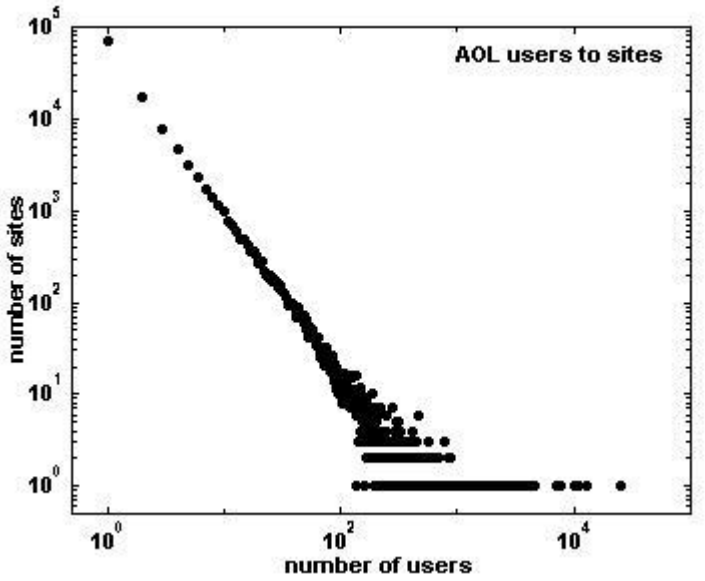
- It is a 'small world'
  - Millions of people. Yet, separated by "**six degrees**" of acquaintance relationships
  - Popularized by Milgram's famous experiment
- Mathematically
  - Diameter of graph is small (log *N*) as compared to overall size
  - Property seems interesting given 'sparse' nature of graph but … This property is 'natural' in 'pure' random graphs

Huge graph with small distance

Web特性→ Web图特性
无标度网络

- 站点大小Site Sizes（以页面数量计算）服从 power law 分布
  - 跨越不同的规模
  - g 在1.6-1.9之间
- 节点的度connections per node 服从power law 分布
  - Study at Notre Dame University reported
  - g = 2.45 for outdegree distribution
  - g = 2.1 for indegree distribution





连接服务器

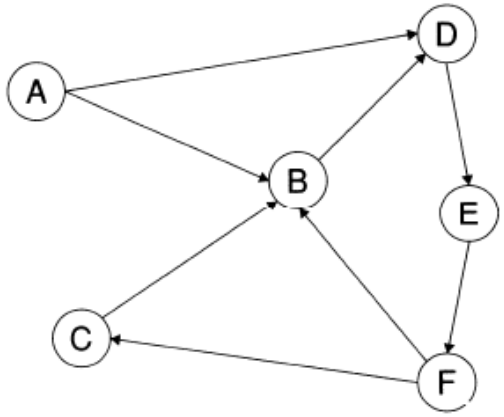
Web图在计算机中如何表示？

- 支持Web图上的快速查询
  - 哪些URL指向给定的URL？
  - 给定的URL指向哪些URL？

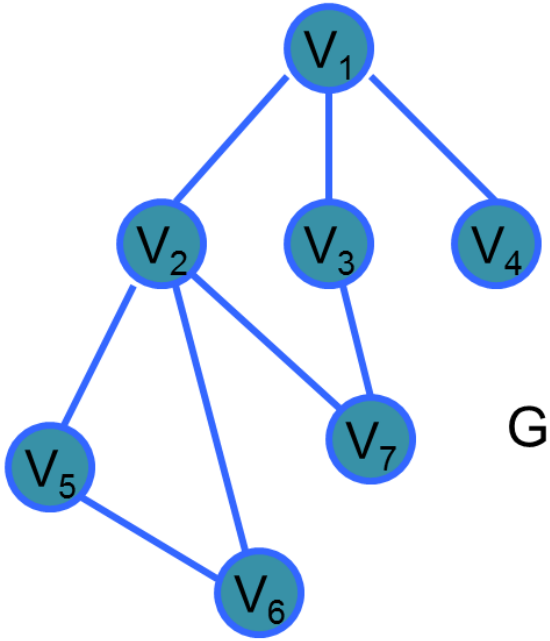
在内存中存储了映射表

  - URL到出链，URL到入链

- 应用
  - 采集控制
  - Web图分析
    - 连通性Connectivity，采集优化
  - 链接分析Link analysis



邻接矩阵 (Adjacency Matrix)



存放顶点的数组

G.vexs=

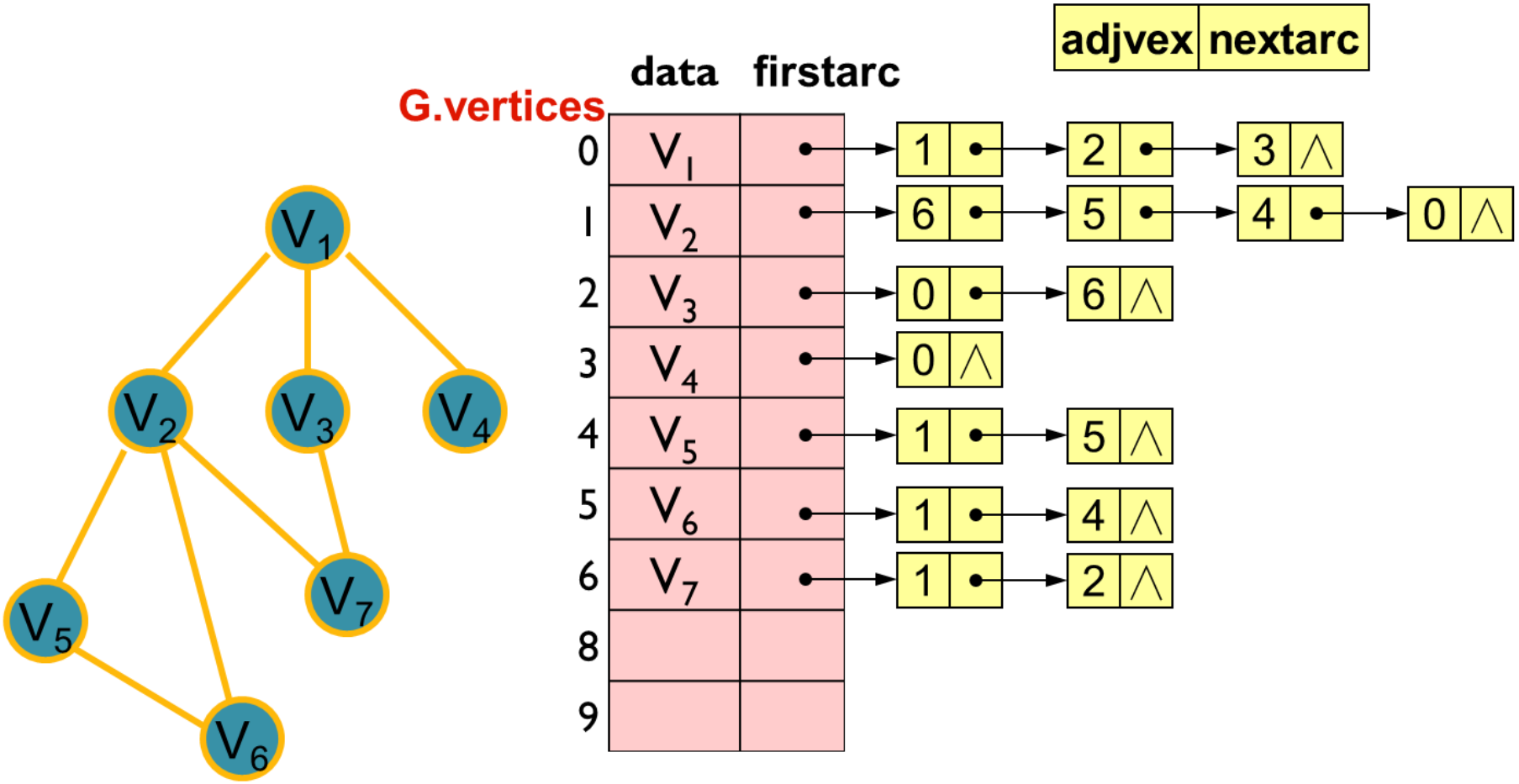
V_1	V_2	V_3	V_4	V_5	V_6	V_7
0	1	2	3	4	5	6

G.arcs=

	0	1	2	3	4	5	6
0	0	1	1	1	0	0	0
1	1	0	0	0	1	1	1
2	1	0	0	0	0	0	1
3	1	0	0	0	0	0	0
4	0	1	0	0	0	1	0
5	0	1	0	0	1	0	0
6	0	1	1	0	0	0	0

表示边的矩阵

邻接表（Adjacency list）



邻接表 vs. 倒排索引

- 假定每个网页都用唯一的整数来表示。我们建立一个类似于倒排索引的邻接表（adjacency table），其每行都对应一个网页，并按照其对应的整数大小来排序。任一网页p 对应的行中包含的也是一系列整数的排序结果，每个整数对应的是链向p 的网页编号。这张邻接表允许我们应答类似于“ 哪些网页指向p？”（“ which pages link top?”）的查询。以同样的方法，可以建立所有p 所指向的网页的邻接表。
- E. g.， 对40亿网页，每个节点需要32bit
  - $\text{Log}_2 4,000,000,000 = 31.897352853986$

邻接表压缩

- 压缩时可以利用的属性：
  - 相似度（表中的很多行有公共元素）
  - 局部性(很多链接会指向很近的网页——同一主机的网页)
  - 在排序表中使用间隔编码

- Eg. Boldi/Vigna 对每个链接只需要3bit进行编码

Boldi/Vigna的思想

- 按照词典顺序对所有URL进行排序，按照排序位置设定ID，生成邻接表如图20-6
  - 1: www.stanford.edu/alchemy
  - 2: www.stanford.edu/biology
  - 3: www.stanford.edu/biology/plant
  - 4: www.stanford.edu/biology/plant/copyright
  - 5: www.stanford.edu/biology/plant/people
  - 6: www.stanford.edu/chemistry

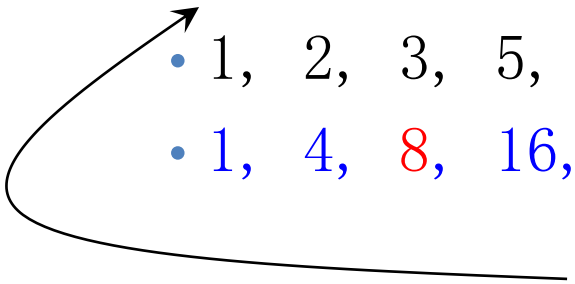
p313书上漏印了图20-6→

```
1: 1, 2, 4, 8, 16, 32, 64
2: 1, 4, 9, 16, 25, 36, 49, 64
3: 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144
4: 1, 4, 8, 16, 25, 36, 49, 64
```

► **Figure 20.6** A four-row segment of the table of links.

Boldi/Vigna

- 这些URL每个都有对应的邻接表
- 主要思想：由于网站的结构化，一个节点的邻接表很可能与字典序的前7个已经处理过的节点相似
- 然后用其中的一个来表示当前邻接表
- 例如：
  - 1, 2, 4, 8, 16, 32, 64
  - 1, 4, 9, 16, 25, 36, 49, 64
  - 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144
  - 1, 4, 8, 16, 25, 36, 49, 64



Encode as (-2), remove 9, add 8

间隔编码

- 使用x，y−x，z−y，…来表示排序好的整数x，y，z，…
- 由于文档间的间隔可能比较紧，采用间隔编码可以进一步减少存储空间

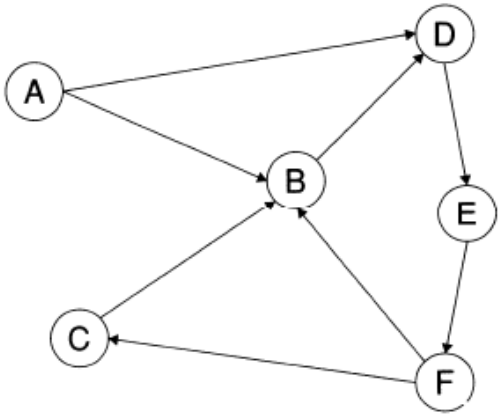
- 回忆：索引压缩也用了间隔编码

Boldi/Vigna的主要优点

- 在规范的排序下仅仅利用局部性
  - 对于互联网，词典序很有效
- 邻接查询能被有效的响应
  - 获取邻接的点时，追溯寻找原型(Protogype)
  - 不需要追溯太远，因为相似性大都是站内的
  - 也可以明确的指定追溯的距离
- 执行一次的算法，不需要迭代

小结：连接服务器

- 哪些URL指向给定的URL？
- 给定的URL指向哪些URL？



- 类似于倒排索引的邻接表
- 邻接表压缩
  - 相似度（表中的很多行有公共元素）
  - 局部性(很多链接会指向很近的网页——同一主机的网页)
  - 在排序表中使用间隔编码

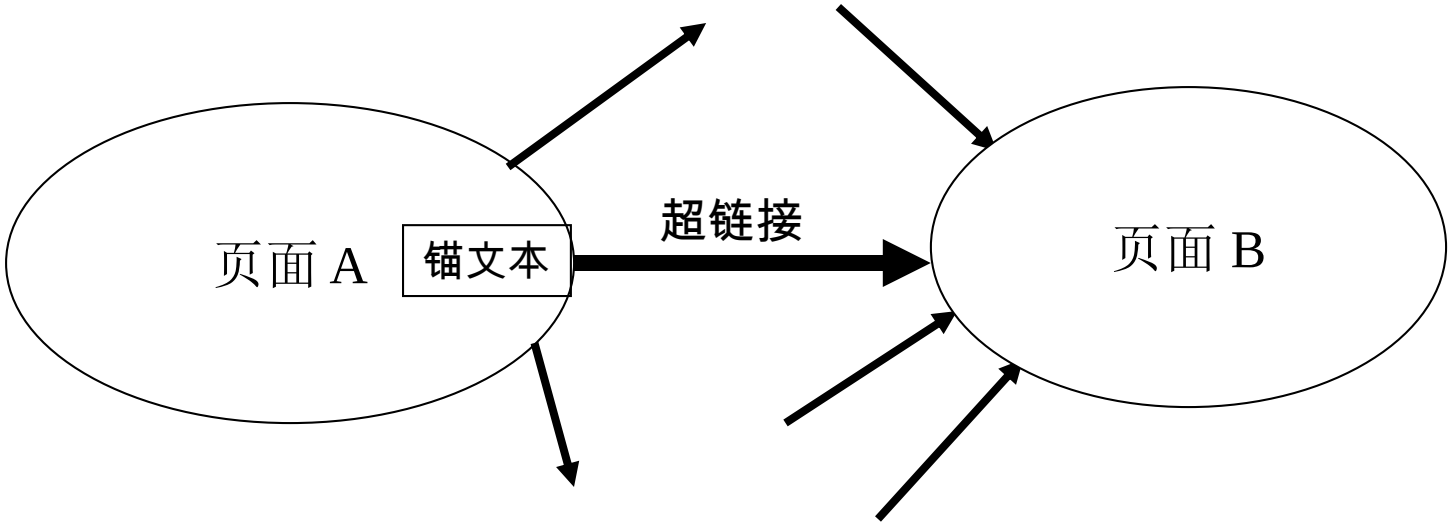
本讲内容：Web搜索

- Web采集
  - 采集器
  - 连接服务器
- 链接分析
  - 锚文本
  - 链接分析：Pagerank
  - 链接分析：HITS

Web上节点（网页）
重要度如何度量？

Web是有向图

文献引用←→入链接



假设 1: A到B的超链接表示A的作者对B的认可

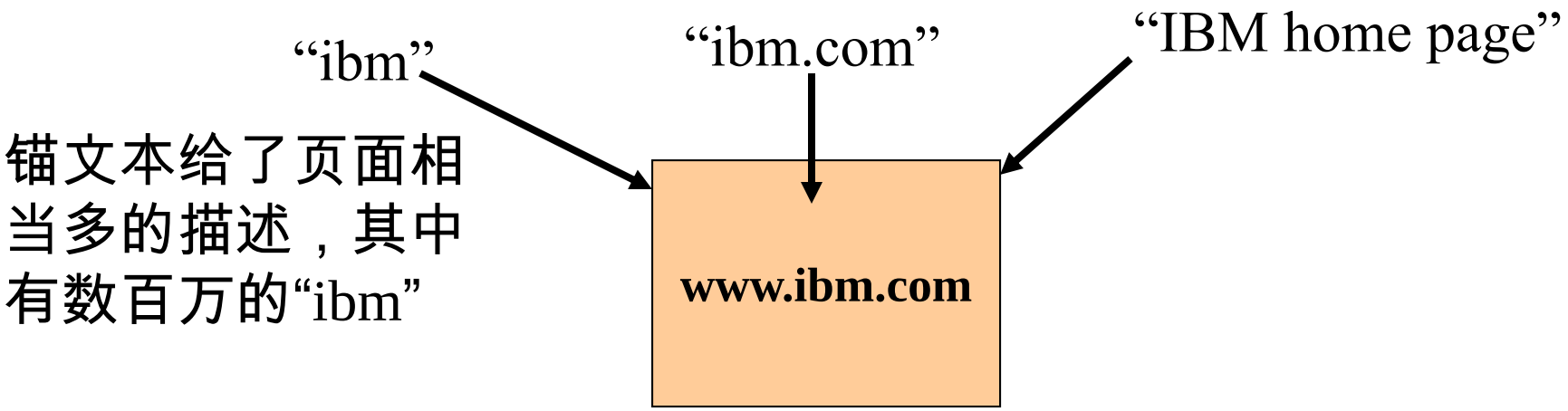
假设 2: 指向页面B的锚文本是对B的一个很好的描述

直觉上上述两个假设是很合理的，但是有特例
例如：大家独用Discuz做论坛，论坛模板？

锚文本

超链接周围还有一些文本，这些文本通常被嵌在<a>标签（称为锚）的href 属性中

- 对于*ibm* 如何在如下三者间进行辨别:
  - IBM's 主页 (基本上都是图片)
  - IBM's 版权声明页 ('ibm' 词频很高)
  - 竞争对手的垃圾信息页面 (任意高词频)

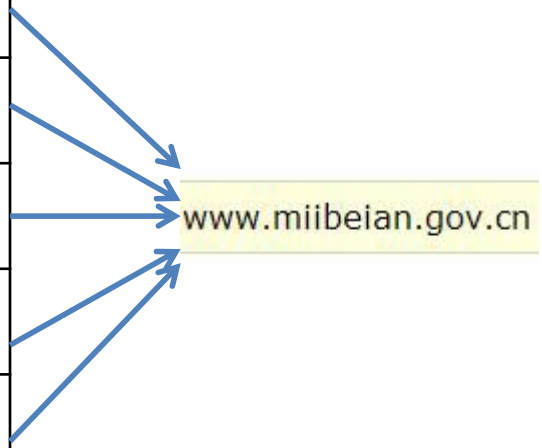


锚文本

- 中文网站锚文本举例
- 雷暴（Thunderstorms）是伴有雷击和闪电的局地对流性天气。

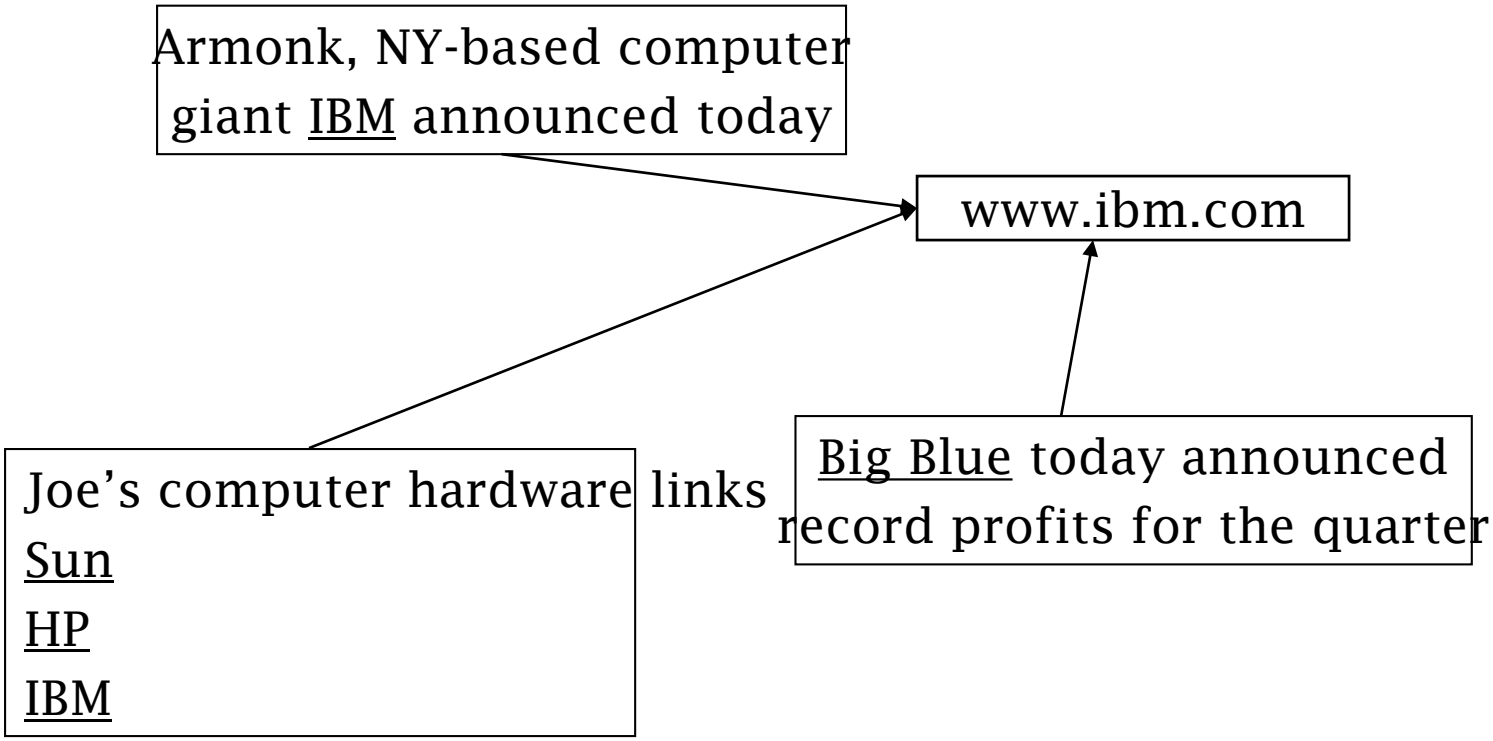
雷击

Sohu.com	京ICP证030367号
Baidu.com	京ICP证030173号
g.cn	ICP证合字B2-20070004号
cntv.cn	京ICP证060535号
xinhuanet.com	京ICP证010042号



索引锚文本

- 在索引文档D的时候，也索引指向文档D的锚文本。



索引锚文本

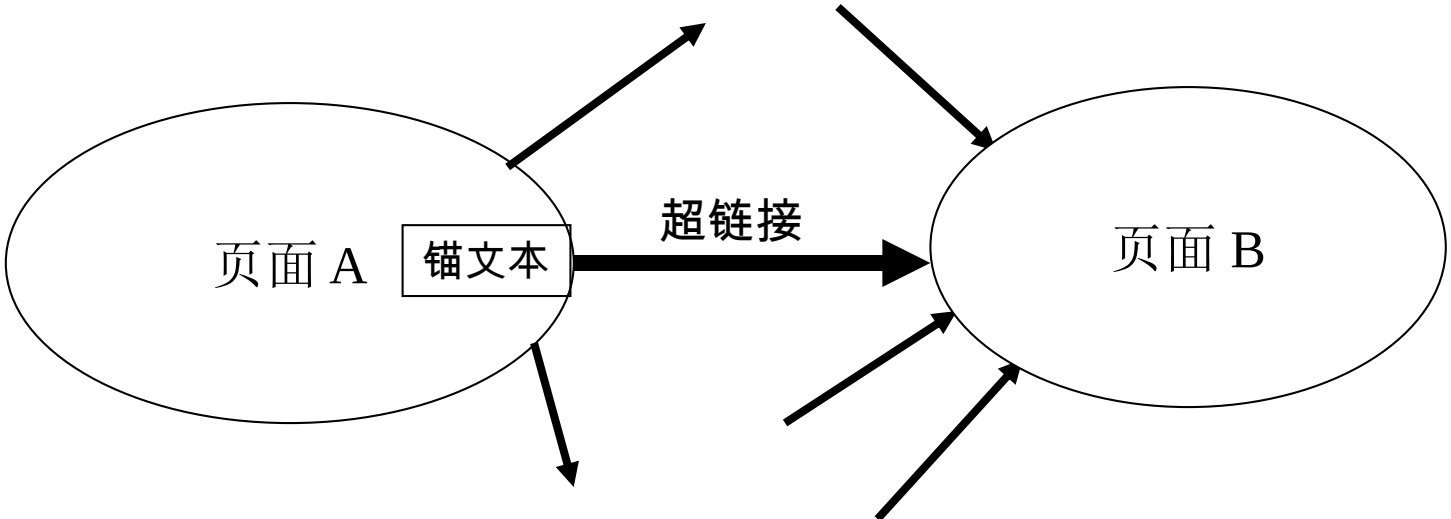
- 锚文本的使用有时候会产生一些有趣的副作用 - *e.g.*, Big Blue
  - 搜索Big Blue时会出现IBM主页，但是主页里面是没有Big Blue这个词的，出现的原因是很多人提到IBM的时候会使用这一绰号
- 可以根据锚文本所在页面的权威性来确定锚文本的权重
  - E.g., 我们认为 cnn.com 和 yahoo.com 的内容是权威的，然后我们就相信它们的锚文本

锚文本

- 其它应用
  - Web图中的链接权重或过滤的依据
    - HITS [Chak98], Hilltop [Bhar01]
  - 用于生成页面的描述文本
    - [Amit98, Amit00]

小结：锚文本

- Web上随处可见的一个现象是，很多网页的内容并不包含对自身的精确描述。
- 因此，Web 搜索者不一定要使用网页中的词项来对网页进行查询，而使用锚文本。
- 锚文本周围窗口中的文本（extended anchor text）也可以当成锚文本一样来使用。

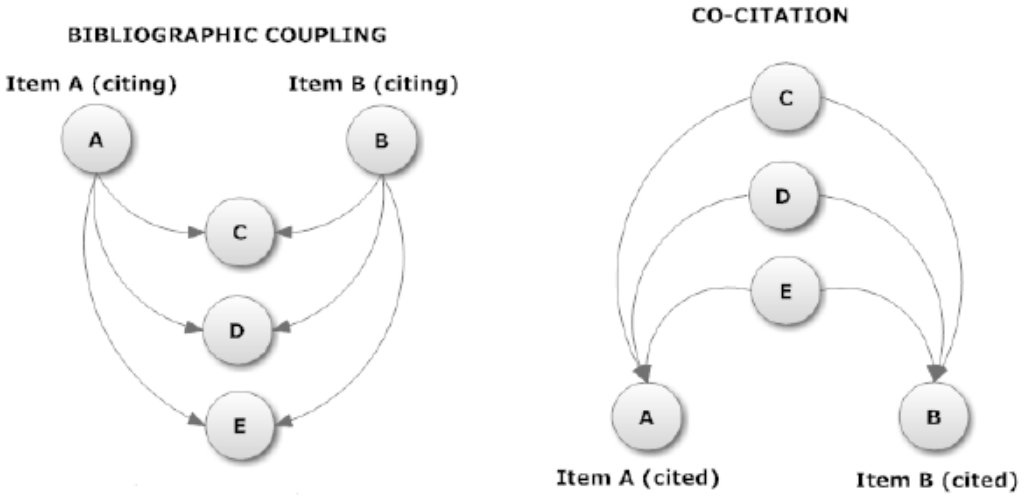


本讲内容：Web搜索

- Web采集
  - 采集器
  - 连接服务器
- 链接分析
  - 锚文本
  - 链接分析：Pagerank
  - 链接分析：HITS

中国PageRank最高的网站？

引文分析

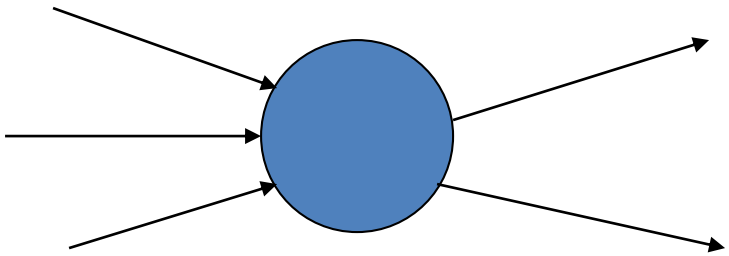


- 引用的频率
- 共引耦合频率(Co-citation coupling frequency)
  - 给作者不同权重后的共引
- 书目耦合频率(Bibliographic coupling frequency)
  - 共同引用了同一篇文章的两篇文章应该是相关的
- 引用索引(Citation indexing)
  - 这位作者被哪些人引用过？ (Garfield 1972)

Pagerank 预览: Pinsker and Narin 1976年提出了期刊权重影响概念，定义与PageRank非常相似

PageRank：对Web图中的每个节点赋一个0-1间的分值查询词无关的排序

- 第一代版本：使用链接的数目作为流行程度的最简单度量
- 两个基本的改进建议:
  - 无向流行度:
    - 赋予每个页面一个分数： 即出链数 + 入链数 (3+2=5).
  - 有向流行度:
    - 页面分数 = 入链数 (3).



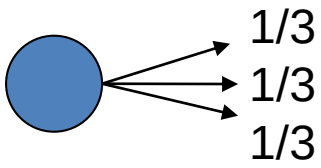
查询处理

- 先检索出所有满足文本查询词的页面 (例如 *venture capital，VC，互联网行业非常重要的给新网站投钱的*).
- 然后把这些页面按照链接的流行度进行排序 (前页的两种计算方法).
- 更复杂的 – 把链接流行度当作静态得分，结合文本匹配的分数进行综合排序（如第七章内容所述）

简单流行度的作弊
Spamming simple popularity

- 思考: 在如下两种计算方式下你怎么作弊能使你的网站得分更高?
  - 无向流行度:
    - 页面分数= 出链数 + 入链数
  - 有向流行度:
    - 页面分数 = 入链数

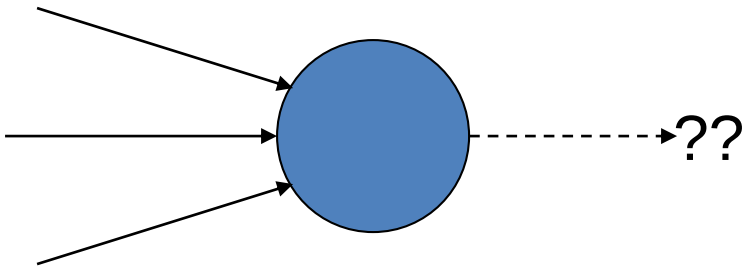
Pagerank打分



- 假设一个浏览者在网络上随机行走:
  - 从一个随机的页面开始
  - 每一步从当前页等概率地选择一个链接，进入链接所在页面的一个
- 在稳定状态下，每个页面都有一个访问概率 – 用这个概率作为页面的分数

尚有缺陷

- 互联网上有很多Dead End
  - Dead End即网页不存在出链，那该怎么办？
  - 这样就没法计算长期游走情况下的访问概率了！



随机跳转（Teleporting）

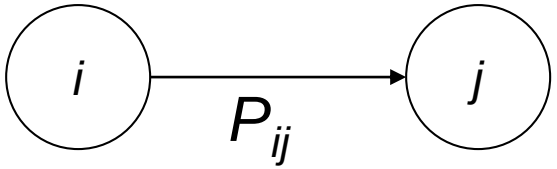
- 遇到dead end时随机跳转到一个页面
- 在非dead end 以 10% 的概率跳转到一个随机页面
  - 以剩余概率从出链中选择一个
  - 10% - 参数，也可以设置成其它值
- 随机跳转的结果
  - 不会再困在一个地方
  - 将会有一个比率表示所有网页在长期的情况下被访问的概率

那么怎么计算这个概率呢？

Markov链

$$P = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,j} & \cdots \\ p_{2,1} & p_{2,2} & \cdots & p_{2,j} & \cdots \\ \vdots & \vdots & \ddots & \vdots & \ddots \\ p_{i,1} & p_{i,2} & \cdots & p_{i,j} & \cdots \\ \vdots & \vdots & \ddots & \vdots & \ddots \end{pmatrix}$$

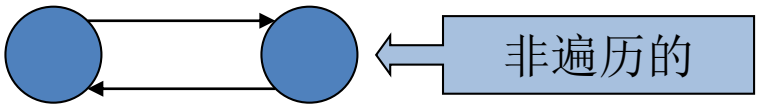
- 一个Markov链有 n 个状态, 以及一个 $n \times n$ 的转移概率矩阵 P.
- 每一步，只能处在一个状态
- $1 \le i,j \le n$,转移概率矩阵的P_{ij} 给出了从状态i到下一个状态j的条件转移概率



- 任意i有 $\sum_{j=1}^{n} P_{ij} = 1.$
- Markov链是随机游走在数学上的抽象

满足该性质的非负矩阵被称为随机矩阵（stochastic matrix）。随机矩阵的一个重要性质是，它的最大特征值是1，与该特征值对应的有一个主左特征向量（principal left eigenvector）。

遍历Markov链



- 满足遍历性的必要条件：
  - 不可约（irreducibility），任意两个状态之间都存在非零概率转移序列
  - 对任意的初始状态，经过有限时间T_0的跳转后，在$T>T_0$时刻处于其它任意状态的概率都大于0
  - 非周期性：不存在两个状态子集之间的循环往复
- 对任意的遍历Markov链，每一个状态都有一个唯一的长期访问率
  - 稳态概率分布
  - 该访问率是与起点无关的

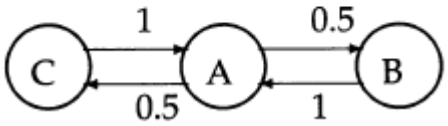
概率向量

- 马尔科夫链的状态概率分布可以看成一个概率向量（probability vector），其中的每个元素都在[0,1]之间，并且所有的元素之和为1。如果一个N 维的概率向量的每个分量对应马尔科夫链中的一个状态的话，那么该向量就可以被看成是在状态上的一个概率分布。
- 将Web图上的一个随机冲浪过程看成是马尔科夫链，其中马尔科夫链中的每个状态对应一个网页，而每个转移概率代表从一个网页跳转到另外一个网页的概率。

$$\begin{pmatrix} 0 & 0.5 & 0.5 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

- 一个概率（行）向量 x = $(x_1, \ldots x_n)$
- 告诉我们随机游走到哪一个地方

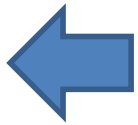
$$\sum_{i=1}^{n} x_i = 1.$$

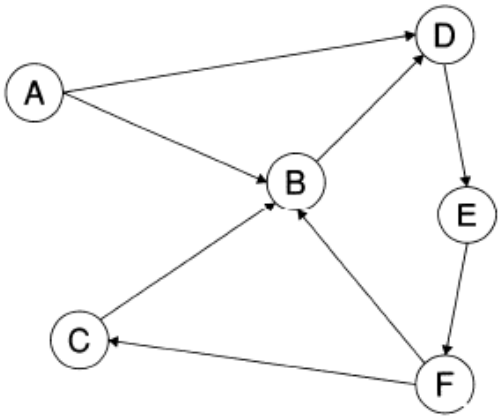


邻接矩阵A→概率转移矩阵P

- Web图的邻接矩阵A可以如下定义：如果存在网页i到网页j的一条链接，那么$A_{ij}=1$，否则 $A_{ij}=0$。

	A	B	C	D	E	F
A	0	1	0	1	0	0
B	0	0	0	1	0	0
C	0	1	0	0	0	0
D	0	0	0	0	1	0
E	0	0	0	0	0	1
F	0	1	1	0	0	0



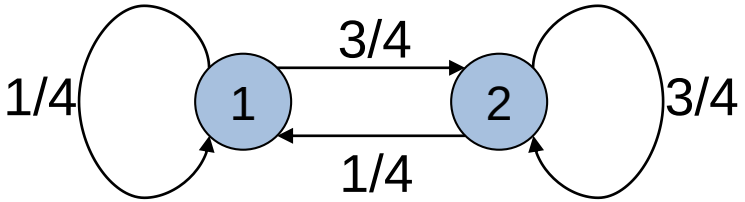


- 如果A的某一行没有1，则用1/N代替每个元素。其他行的处理如下：
- (1) 用每行中的1 的个数去除每个1，因此如果某行有3 个1，则每个1 用1/3 代替；
- (2) 上面处理后的结果矩阵乘以1−α；
- (3) 对上面得到的矩阵中的每个元素都加上α/N。

概率向量的变化

- 在当前这一步的概率向量是 x = $(x_1, \ldots x_n)$ 那么下一步的概率向量是多少?
- 回想一下，转移概率矩阵P告诉我们在状态i如何转移到其他状态
- 下一步的概率向量就是 xP.

稳态概率



- 稳态看起来就像一个概率向量 a = $(a_1, \ldots a_n)$:
  - a_i 就是我们在状态 i 的概率
  - **在上面这个例子中,** $a_1=1/4$ and $a_2=3/4$.
- 怎么计算稳态概率？
  - 假设 a = $(a_1, \ldots a_n)$ 表示稳态概率
  - 如果我们当前状态是a，那么下一步的分布应该是 aP
  - 因为已经是稳态了，所以 a=aP
    - 解矩阵等式可以得到 a.
    - a是P 的主左特征向量(对应于P最大特征值的特征向量)
    - 转移概率矩阵的最大特征值是1

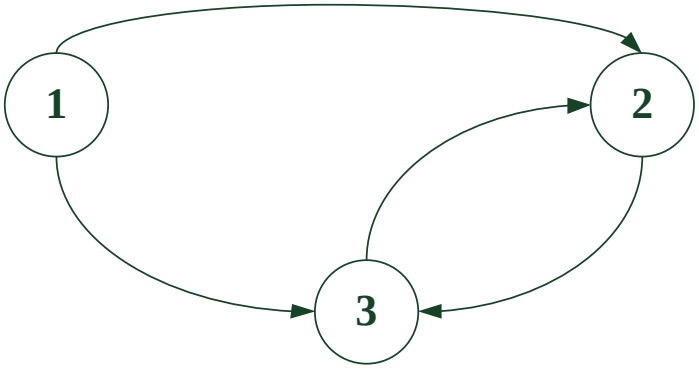
计算a的一种方法：幂迭代(power iteration)

- 回忆一下，不管我们从什么地方开始，最终都会到达稳定状态 a.
- 从任意分布开始 (例如 x=(*10…0*)).
- 一步之后，到达 xP;
- 两步之后 xP^2 , 然后 xP^3 ， ….
- “最终” 意味着当*k*很大时, $xP^k = a$.
- 算法: 给 x 乘上 P 的k次方，k不断增加，直到乘积看起来已经稳定.

PageRank计算示例习题21-6

从模型到求解：Web图→邻接矩阵A

- 考虑一个具有3个节点（ID分别为1、2、3）的Web图。其中的连接为：1→2、1→3、2→3、3→2。



$$A = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

- 邻接矩阵：A_{ij}为1表示网页i指向网页j，为0表示网页i没有指向网页j

PageRank计算示例习题21-6
邻接矩阵A→概率转移矩阵P

- 随机跳转时参数α=0.1，求A对应的概率转移矩阵P

$$\alpha = 0.1,\ N = 3 \Rightarrow (1-\alpha) = \frac{9}{10},\quad \frac{\alpha}{N} = \frac{1}{30}$$

$$A = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \xrightarrow{\text{归一化}} \Rightarrow \begin{bmatrix} 0 & 1/2 & 1/2 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \xrightarrow{\times(1-\alpha)} \Rightarrow \begin{bmatrix} 0 & 9/20 & 9/20 \\ 0 & 0 & 9/10 \\ 0 & 9/10 & 0 \end{bmatrix}$$

$$\xrightarrow{+\frac{\alpha}{N}} \Rightarrow \begin{bmatrix} 1/30 & 29/60 & 29/60 \\ 1/30 & 1/30 & 28/30 \\ 1/30 & 28/30 & 1/30 \end{bmatrix} = \frac{1}{60}\begin{bmatrix} 2 & 29 & 29 \\ 2 & 2 & 56 \\ 2 & 56 & 2 \end{bmatrix}$$

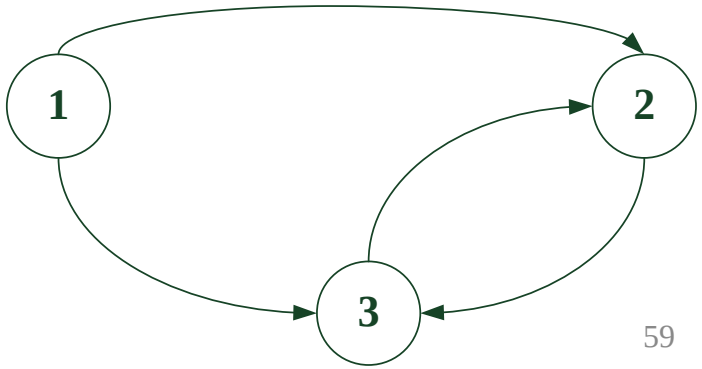
$$\Rightarrow \quad P = \frac{1}{60}\begin{bmatrix} 2 & 29 & 29 \\ 2 & 2 & 56 \\ 2 & 56 & 2 \end{bmatrix}$$

PageRank计算示例习题21-6

求P最大特征值对应的特征向量

$$[x_1 \quad x_2 \quad x_3]\cdot\begin{bmatrix}2 & 29 & 29\\ 2 & 2 & 56\\ 2 & 56 & 2\end{bmatrix}=60\cdot[x_1 \quad x_2 \quad x_3]$$

$$\begin{cases}2x_1+2x_2+2x_3=60x_1\\ 29x_1+2x_2+56x_3=60x_2\\ 29x_1+56x_2+2x_3=60x_3\end{cases}\Rightarrow\begin{cases}4x_2=58x_1\\ x_2=x_3\end{cases}\Rightarrow\begin{cases}x_1=1/30\\ x_2=29/60\\ x_3=29/60\end{cases}$$



PageRank

PageRank计算示例 例21-1

一个小型Web 图

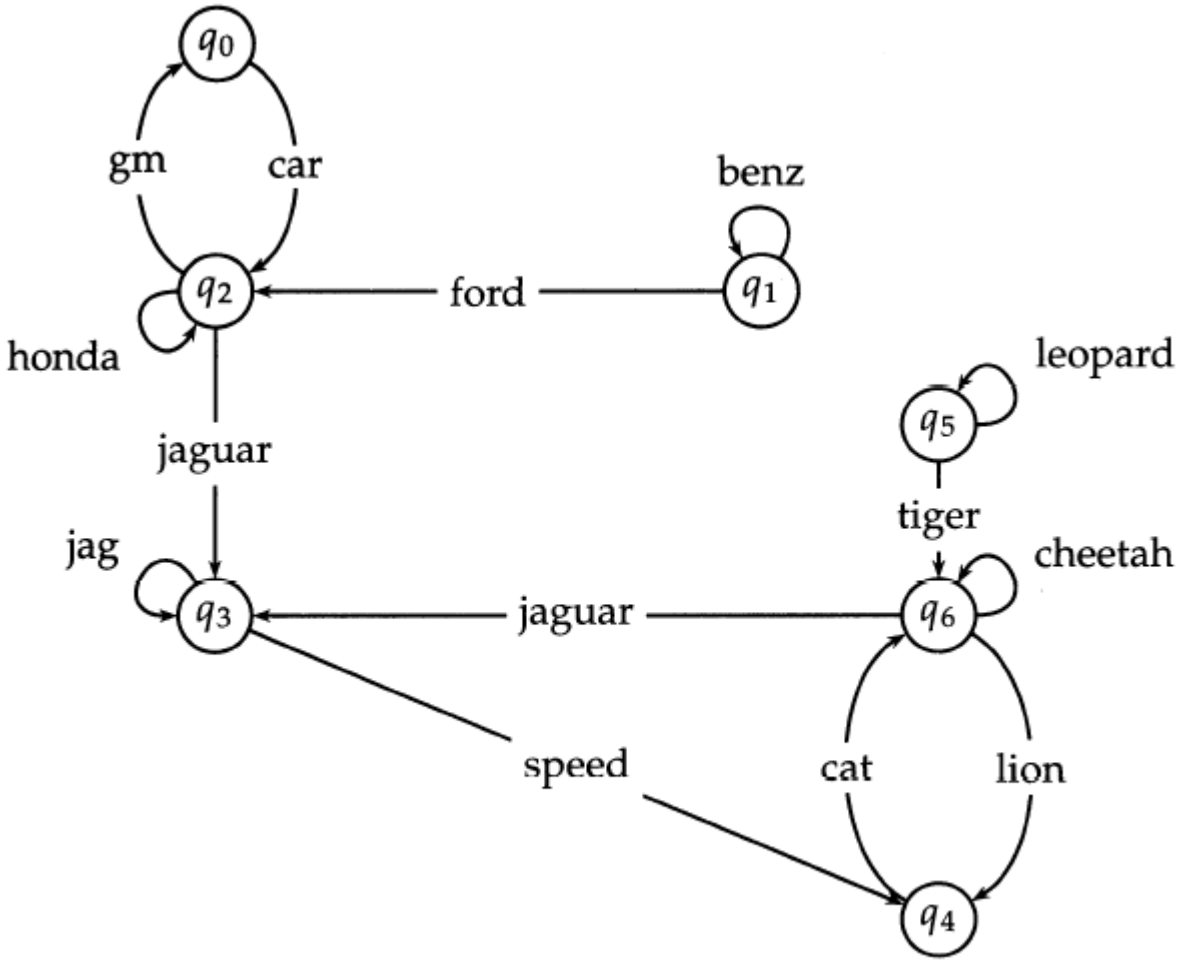


图 21-4 一个小型 Web 图。每条边上都用链接上的锚文本单词来标记

PageRank计算示例 例21-1

一个小型Web 图

假定随机跳转操作的概率是0.14，其（随机）转移概率矩阵为：

0.02	0.02	0.88	0.02	0.02	0.02	0.02
0.02	0.45	0.45	0.02	0.02	0.02	0.02
0.31	0.02	0.31	0.31	0.02	0.02	0.02
0.02	0.02	0.02	0.45	0.45	0.02	0.02
0.02	0.02	0.02	0.02	0.02	0.02	0.88
0.02	0.02	0.02	0.02	0.02	0.45	0.45
0.02	0.02	0.02	0.31	0.31	0.02	0.31

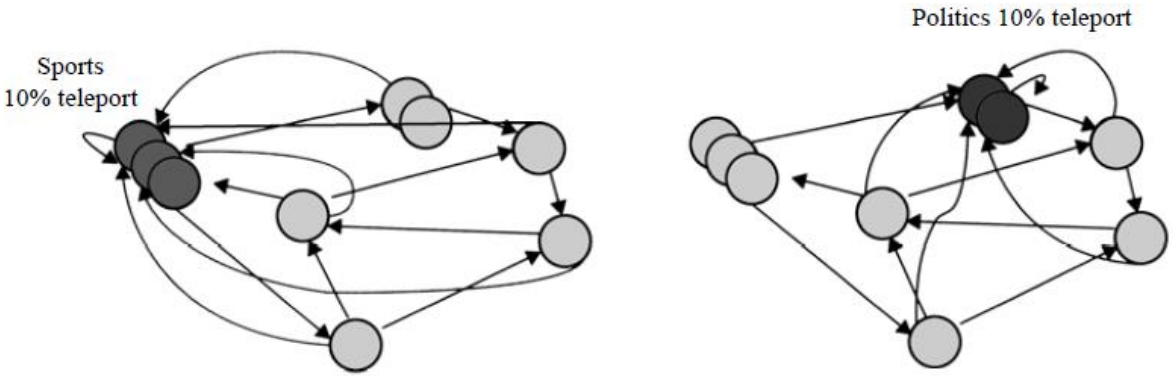
该矩阵的PageRank 向量为：

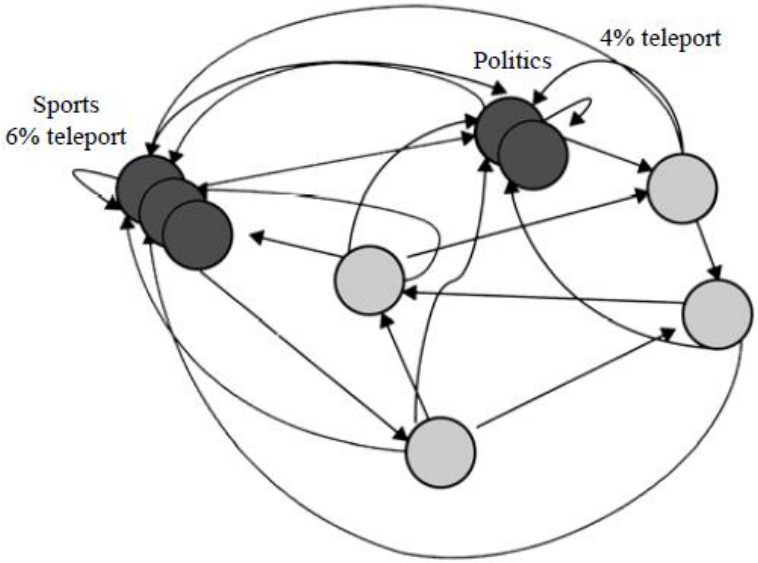
$$\vec{x} = (0.05 \quad 0.04 \quad 0.11 \quad 0.25 \quad 0.21 \quad 0.04 \quad 0.31)$$

PageRank小结

- 预处理:
  - Web图→邻接矩阵A→概率转移矩阵 **P** ←p324,习题21-6
  - 由P计算 **a**
  - 元素 a_i 是一个0和1之间的数: 即页面 i 的PageRank
- 查询处理:
  - 检索满足查询要求的页面
  - 按PageRank排序
  - 排序与查询词是无关的

面向主题的Pagerank





考虑一个体育兴趣为60%、政治兴趣为40%的用户。如果随机跳转操作的概率是10%的话，那么其中有6%是到体育类，而4%到政治类网页

PageRank 都可以被表示成多个面向主题的PageRank 的线性组合。

$$0.6\vec{\pi}_s + 0.4\vec{\pi}_p$$

$\vec{\pi}_s$ 和 $\vec{\pi}_p$ 分别是面向体育和政治主题的 PageRank 向量

本讲内容：Web搜索

- Web采集
  - 采集器
  - 连接服务器
- 链接分析
  - 锚文本
  - 链接分析：Pagerank
  - 链接分析：HITS

超链导向的主题搜索
Hyperlink-Induced Topic Search (HITS)

- 对每个网页给出两个得分：一个得分被称为hub值，另外一个被称为authority值

- 作为查询的响应，与排序过的相关网页列表不同，可以找到两个互相联系的页面集合:
  - 导航页*Hub pages* 很好的指向某一主题的列表
    - e.g., “Bob’s list of cancer-related links.”
  - 权威页*Authority pages* 在针对某一主题的好Hub页中经常出现
- 相对于寻找特定页面，更加适合于泛主题搜索
  - E.g., wish to learn about leukemia(白血病)

p325

导航Hubs 和权威Authorities

- 针对某一主题的好Hub页会指向很多关于这个主题的Authority页面
- 关于某一主题的好Authority页面会被很多针对这一主题的好Hub页指向
- 循环定义Circular definition → 可以迭代求解页面的Hub值和Authority值

HITS步骤：确定基本集

- 计算HITS需要一个Web子集
  - 一个包含好的hub 网页和authority 网页的Web 子集及它们之间的链接。
- 获取该Web子集
  - 给定查询词 (如 leukemia)，用文本索引取出所有包含leukemia的所有网页。这些网页称为根集合（root set）。
  - 将根集及指向根集中的网页和根集所指向的网页加入到基本集（base set）中。
- 之所以如此构造基本集的原因在于：
  - 一个好的authority 网页可能不包含查询文本
  - 从根集到基本集的扩展能够增加更多好的hub 网页及authority 网页

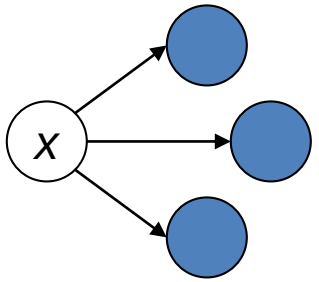
HITS步骤：精选出Hub页和Authority页

- 对于基本集中的每一个页面x计算Hub分*h(x)* 和Authority分 *a(x)*.
- 初始化: 所有的 *x*, *h(x)←1; a(x) ←1;*
- 迭代更新 *h(x), a(x)*;
- 迭代之后
  - 输出具有最高 *h()* 的页面作为Top Hub页
  - 最高 *a()* 的页面作为Top Authority页

HITS步骤：迭代更新

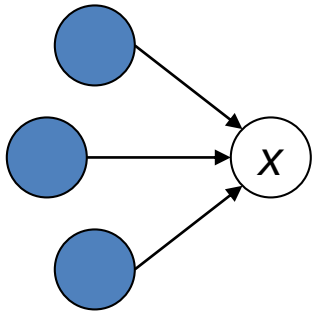
- 对所有 x 重复如下步骤:

$$h(x) \leftarrow \sum_{x \mapsto y} a(y)$$



如果从 v 到 y 存在一条超链接，则记为 $v \mapsto y$

$$a(x) \leftarrow \sum_{y \mapsto x} h(y)$$



- 为了避免 h() 和 a() 太大, 每次迭代之后都可以把它们按一定比例缩小。缩放并不会影响最后结果:因为我们只关心相对的分数

应该迭代多少次？

- 根集包含所有的与文本查询匹配的网页，实施时建议取200 左右的网页就足够了。
- 迭代一些次数后分数会收敛：
  - 我们只需要 h() 和 a() 的相对顺序，而不需要它们的值
- 利用幂迭代算法有可能只需要少量迭代次数就可以获得高hub 值和高authority 值网页的相对次序。
- 实验结果表明，实际上公式（21-8）只需要大概5次迭代就可以产生相当好的结果。

写成矩阵形式

$$h(x) \leftarrow \sum_{x \mapsto y} a(y)$$

$$a(x) \leftarrow \sum_{y \mapsto x} h(y)$$

$n \times n$ 邻接阵 A

- h=Aa.
- a=A^th.

A^t 是 A的转置

替换, h=AA^th a=A^tAa.

那么, h是AA^t 的特征向量 , a是 A^tA的特征向量

那么, 前述迭代算法就是一个特殊的、已知的求解特征向量的算法: *power iteration* 方法.

已被证明是收敛的

HITS计算步骤小结

- 最后的计算形式如下：
- (1) 收集Web 网页构成网页子集，利用网页的链接形成图结构，计算AA^T 及A^TA；
- (2) 计算AA^T 及A^TA 的主特征向量，得到最后的*hub*值向量h和authority值向量a；
- (3) 输入排名靠前的hub 网页和authority 网页。

HITS计算示例 例21-2
一个小型Web 图

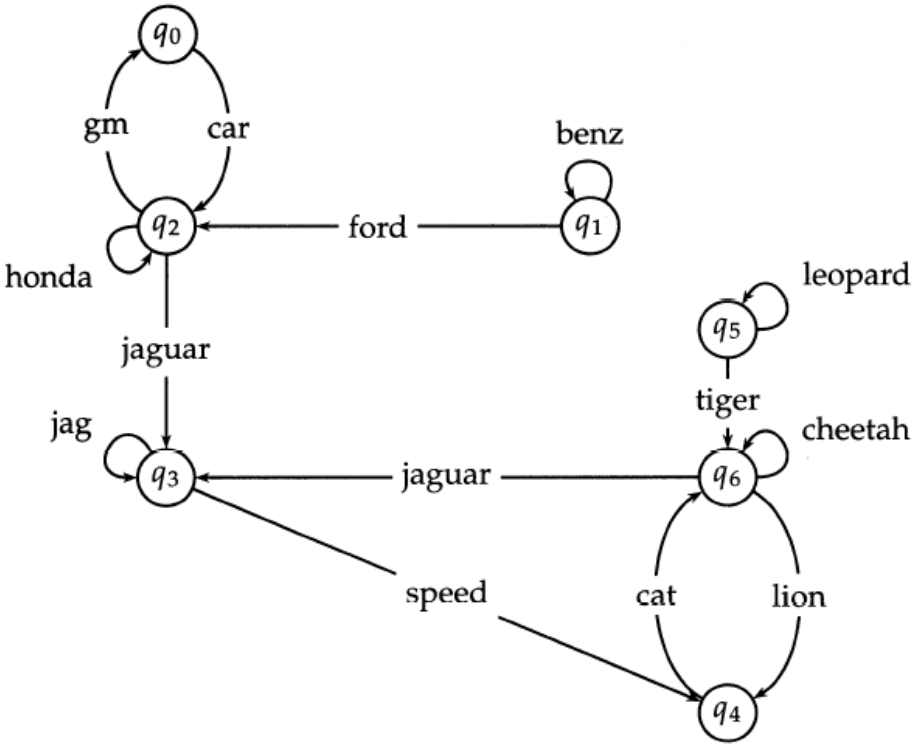


图 21-4　一个小型 Web 图。每条边上都用链接上的锚文本单词来标记

例**21-2** 假定查询为jaguar，我们将锚文本包含该查询词的链接的权重加倍，则图21-4 对应矩阵A 如下：

$$\begin{matrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 2 & 1 & 0 & 1 \end{matrix}$$

$$\vec{h} = (0.03 \quad 0.04 \quad 0.33 \quad 0.18 \quad 0.04 \quad 0.04 \quad 0.35)$$

$$\vec{a} = (0.10 \quad 0.01 \quad 0.12 \quad 0.47 \quad 0.16 \quad 0.01 \quad 0.13)$$

小结：HITS

- 超链导向的主题搜索
  Hyperlink-Induced Topic Search (HITS)

- 对每个网页给出两个得分：一个得分被称为hub值，另外一个被称为authority值
- HITS步骤
  - 确定基本集
  - 精选出Hub页和Authority页
  - 迭代更新

$$h(x) \leftarrow \sum_{x \mapsto y} a(y)$$

$$a(x) \leftarrow \sum_{y \mapsto x} h(y)$$

- h是AA^t 的特征向量，a是 A^tA的特征向量

今日小结

- Web采集
  - 采集器基本结构
  - Web图→邻接矩阵、邻接表
- 链接分析
  - Pagerank：A→P→a
  - HITS：h是AA^t 的特征向量，a是 A^tA的特征向量