基于后缀树的互联网信息检索

福建师大附中　吴连龙

wll@wll.name

http://www.mydrs.org

提　　纲

- 互联网信息服务现状
- 信息查询系统的设计
- 后缀树与后缀数组算法
- 信息检索系统的实现
- 对等网信息查询系统
- 结束语

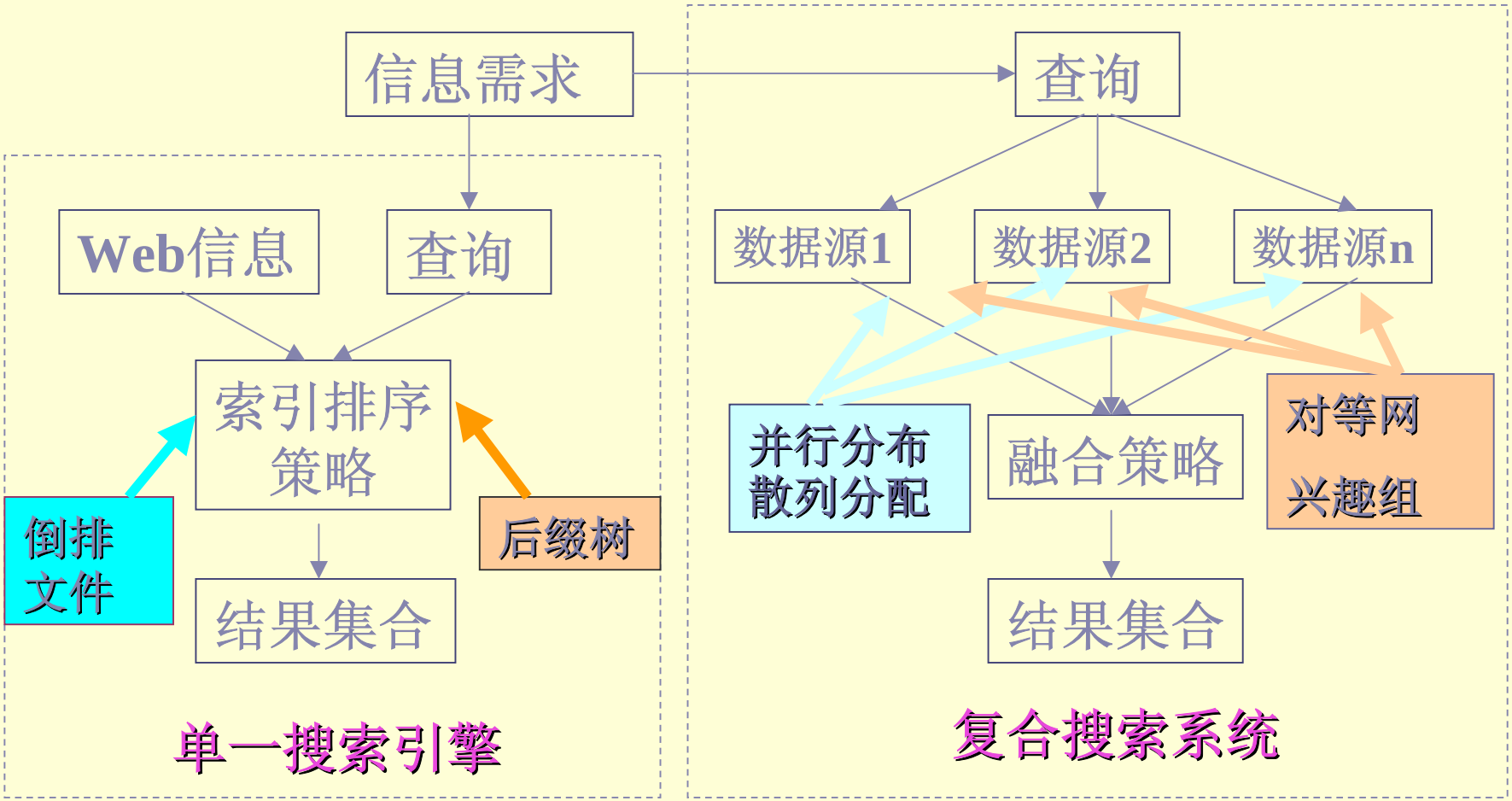
互联网信息服务现状

- 互联网信息飞速增长
  - 中国www站点数超过62万个
  - 上网用户总数达8700万人 （CNNIC，2004）
- 网络信息服务面临挑战
  - 挑战1 反应速度
    - 迅速从海量信息中获得指定信息
    - 及时跟踪信息的动态变化
  - 挑战2 用户需求
    - 查找特定的主页或网页(点)
    - 获得同主题的相关站点(面)
  - 挑战3 自主个性
    - 主动向用户推送信息
    - 提供个性化信息服务

搜索引擎技术

- 网络信息搜索技术
  - 国际：在ACM 学报和论文集中有许多这方面的文章, Arasu 等发表在ACM Transactions on Internet Technology (2001,v1,No1)上题为“Searching the Web”的论文提出搜索引擎的主要技术：网页抓取技术,web 信息挖掘技术,存储与索引技术,搜索与查询技术,查询结果评价技术等。
  - 国内： 北大天网、华工木棉；百度、中搜、一搜、搜狗
- 倒排文件机制
  - 面向单词，建立在语言词库的基础上
  - 拆分短语，进行复杂的集合运算
  - 查询结果的准确性和完整性依赖于中文分词的效果

信息查询系统的设计框架



后缀树的研究与应用

- 后缀树的出现
  - Weiner 于1973年提出，McCreight 在1976年和 Ukkonen 在1992, 1995年进一步完善算法。
  - E. M. McCreight. A Space-economical Suffix Tree Construction Algorithm. J. ACM, 23(2):262--272, 1976
- 后缀树的应用
  - 字符串处理
  - DNA序列比对
  - 文本聚类
  - XML结构索引

后缀树的实例

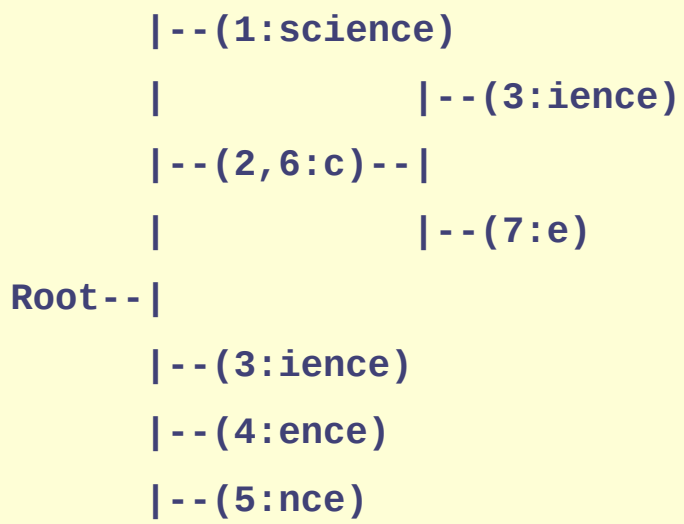
字符串“science”的7个后缀字符串分别是：

Suffix (1) = science
Suffix (2) = cience
Suffix (3) = ience
Suffix (4) = ence
Suffix (5) = nce
Suffix (6) = ce
Suffix (7) = e

后缀字符串按字典顺序排序后的结果是：

Suffix (6) = ce
Suffix (2) = cience
Suffix (7) = e
Suffix (4) = ence
Suffix (3) = ience
Suffix (5) = nce
Suffix (1) = science

对字符串“science”建立的后缀树如下：



后缀树构建后，不仅使字符串更加紧凑，还可以高效地实现比如子串查找、最长重复子串、最长公共子串、回文子串等众多功能。

后缀数组的数据结构

后缀数组Index，依次存放排序好的后缀字符串的开头位置。
例如“science”的后缀数组为：

Index	1	2	3	4	5	6	7
Value	6	2	7	4	3	5	1

名次数组 Rank ， Rank[i]存放Suffix (i)在排序中的名次。
例如“science”的名次数组为：

Rank	1	2	3	4	5	6	7
Value	7	2	5	4	6	1	3

为了获得后缀数组，显而易见的方法是将所有后缀子串看作独立的字符串排序，但这样复杂度太高，不能使人满意。

后缀数组的概念定义

定义1：对于字符串S，定义S的长度为Len(S)，第i个字符为S[i]，第i个字符至第j个字符组成的子串为S[i..j] 。构成字符串的字符集Σ。

定义2：Suffix(i)的k-前缀 = S[i..i+k-1]，即Suffix(i)的前k个字符组成的字符串，如果Len(Suffix(i))<k，则其k-前缀 = Suffix (i)。

定义3：按所有后缀字符串的k-前缀排序的后缀数组为$Index_k$，相应的名次数组为$Rank_k$。

后缀数组的生成算法

- 计算$Index_1$和$Rank_1$数组的算法：
  1. 按1-前缀（即首字母）对所有后缀排序，生成后缀数组$Index_1$。这里可以采用快速排序（Quick Sort），时间复杂度O（nlogn），或者采用桶排序（Bin Sort），时间复杂度O（|Σ|）。
  2. 计算基于1-前缀的名次数组$Rank_1$。允许并列的名次，即对于后缀数组中的第i个后缀（i≥2），如果与第i-1个后缀字符串相等，则名次与第i-1个后缀相同，否则名次等于i。时间复杂度O（n）。

后缀数组的生成算法（续）

- 基于$Index_k$和$Rank_k$的2k-前缀Suffix(x)与Suffix(y) 大小
  - “相等”等价于

    $Rank_k[Index_k[x]] = Rank_k[Index_k[y]]$

    且 $Rank_k[Index_k[x+k]] = Rank_k[Index_k[y+k]]$
  - “小”等价于

    $(Rank_k[Index_k[x]] = Rank_k[Index_k[y]]$ 且 $Rank_k[Index_k[x+k]] < Rank_k[Index_k[y+k]])$

    或 $(Rank_k[Index_k[x]] < Rank_k[Index_k[y]])$

这种比较大小方法的时间复杂度为**O(1)**，而朴素逐个字符比较大小的方法最坏情况需要**O(2k)**的时间。

后缀数组的生成算法（续）

- 算法总框架

  k = 1

  repeat

  k-前缀排序

  计算名次

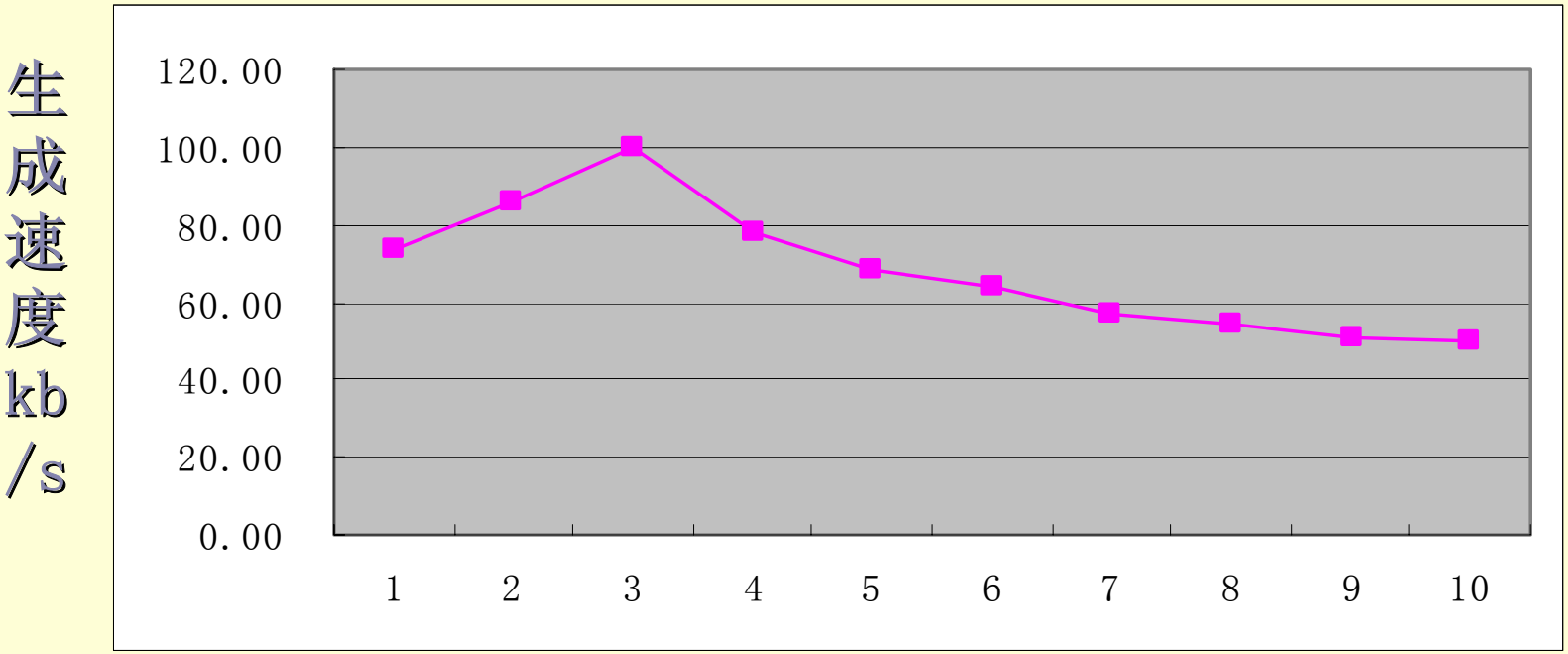
  k = k * 2

  until k >= 短语长度

后缀树与后缀数组的对比

	后缀树	后缀数组
数据结构	树	数组
实现方法	复杂	简洁
可扩展性	较好	很好
时间复杂度	O（n*|Σ|） O（n*log|Σ|）	O（nlogn）

后缀数组算法性能曲线



横坐标表示随机单字节文本长度，以10kb为单位

随着文本的增大，性能的变化趋于平稳。

超文本网页处理

- 提取文本信息
  - 忽略Html标签、Style样式表、Script脚本
  - 内容文本保存为Unicode记录文件
- 建立新的文件体系
  - 分块

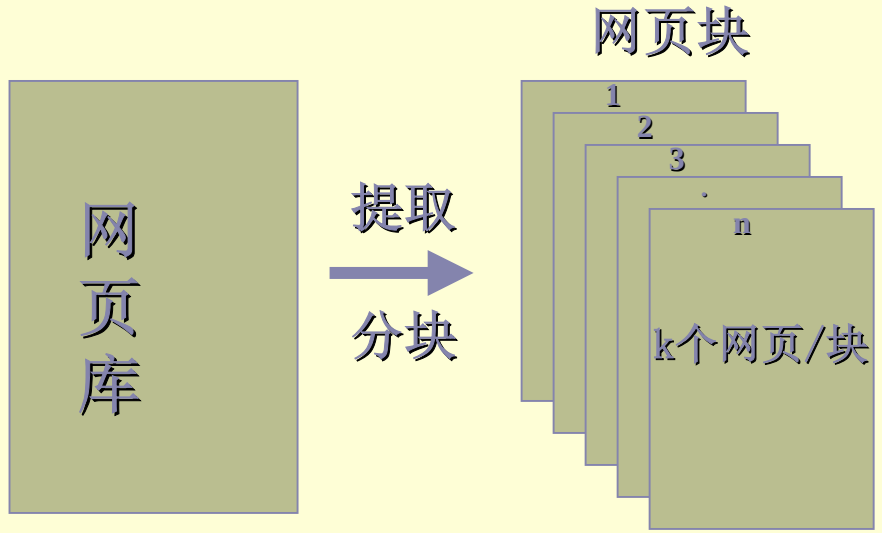
每k个网页合为1块

随机、Hash或文件聚

  - 建立网页属性表

保存网页URL、长度、在相应内容文件中的开始位置。

- 性能



网页属性表

编号	URL	长度	开始位置
1	http-1	len-1	pos-1
2	http-2	len-2	pos-2
…	…	…	…
k	http-k	len-k	pos-k

文本内容到URL的查询

查询 nce

nce

Suffix (6) = ce
Suffix (2) = cience
Suffix (7) = e
Suffix (4) = ence
Suffix (3) = ience
Suffix (5) = nce
Suffix (1) = science

后缀数组

可针对短语、句子甚至文章，在索引中进行高效率的全文查询。这是基于关键字的索引机制难以做到的。

多关键字的包含与不包含查询，最长重复子串、最长公共子串、回文子串、中文分词等功能

5

网页属性表

页面号	URL	长度	开始位置
1	http-1	len-1	pos-1
2	http-2	len-2	pos-2
k	http-k	…	…

http-2

URL等属性

后缀数组搜索的结果输出

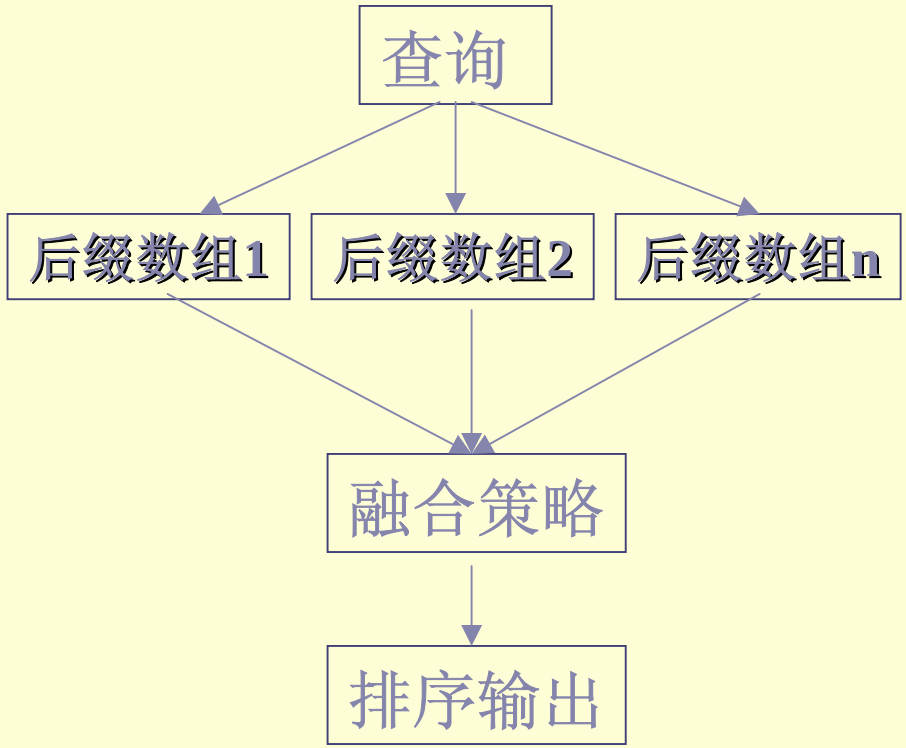
- 融合策略

  根据匹配程度给出一个综合的输出结果表

- 排序输出

  按照查询要求计算相似度排序输出

- 查询模式
  - 大内存
  - 零内存
- 缓存机制



下一步工作:对等网信息查询系统

- **P2P搜索查询**
  每个节点可提供信息可提出和转发查询，没有集中的服务器。
- **泛洪式 P2P查询**
  接到查询信息的节点不断地向所有邻居转发查询信息。
- **漫游式 P2P查询**
  接到查询信息的节点不断地向一个邻居转发查询信息。
- **共同兴趣组P2P查询**
  每个节点（用户）都有自己兴趣信息和有共同兴趣的节点连接（通讯簿）。利用兴趣组通讯簿为所收到的查询提供指南是这种查询算法的核心，它可减少网络负荷并提高速度与精度。

下一步工作:应用新技术

- 人工智能技术

  自然语言处理，WEB数据挖掘，知识发现，机器学习，智能代理，智能推理等。

- 个性化搜索技术

  如：Eric 等,Web Search---Your Way, Communications of the ACM , Volume 44 , Issue 12 (2001) P: 97 – 102

- 自适应命名实体识别技术

  如：Zhu 等 Adaptive Named Entity Recognition（2004）
  http://kmi.open.ac.uk/people/jianhan/ESpotter/

- ……

结 束 语

- 指出网络信息服务面临的三个挑战
- 网络信息搜索系统的设计结构
- 将后缀数组算法应用与信息检索
- 建立相应的查询方法
- 展示下一步的工作
- 希望得到各位专家学者的建议和支持

wll@wll.name

http://www.mydrs.org

谢谢!