**Data Structure of Geeking**

**原始网页存储**

方法一：

以html文件形式存储。

以分类目录方式存储：sina、sohu目录（或basketball/football）等

缺点：构建倒排索引时输入速度慢。

方法二：

按照分类文件存储。

同一类型的文件按照一定格式存储于同一个文件中。

E.g.

**原始网页库中的一条网页记录**

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx // 之前的记录

version:1.0 // 记录头部

url:http://ast.nlsde.buaa.edu.cn/

date:Mon Apr 05 14:22:53 CST 2010

IP:218.241.236.72

length:3981

<!DOCTYPE …… // 记录数据部分

<html> …… </html>

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx // 之后的记录

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

缺点：不能按照网页 URL 直接定位到所指向的网页。

需要建立网页库索引。不过网页库索引可同时存储网页标题、摘要、日期等信息；

法1要存储这些信息也需要建立网页信息数据表，这点上扯平。

**网页信息表**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 文档ID | URL | 标题 | 摘要 | 日期 | 类型 | 偏移(法二) | 关键字 |
| 18 | [www.sina.com](http://www.sina.com)... | XxX | XXX | XX | 篮球 | 128 | X,X,X |

文档ID：主键，（映射为URL，一一对应）。

URL：非常重要

标题：用于 结果显示/分层索引

摘要：用于 结果显示（如果动态摘要就忽略词项）/分层索引

日期：网页日期

类型：网页类型（具体根据什么分类有待商议），用于 法二存储网页/根据类型进入目录读取文件/分类搜索等？

关键字：体育新闻网页上的关键字，用于 结果显示/结果排序

偏移：法二独有

原始文档路径：

法一：根据 类型+url 进入目录读取原始文档

法二：根据 类型+偏移进入文件读取原始文档

两种方法均需要先从配置文件读取目录的 绝对路径

**正向索引**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 文档ID | 词项ID | 词项ID |  |  |  |  |  |
| 1 | 极客 |  |  |  |  |  |  |

注：此处的词项会有重复项。

构建过程中，生成 词项 - 词项ID 的映射，存入数据库

|  |  |
| --- | --- |
| 词项ID | 词项 |
| 1 | 极客 |

**倒排索引**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 词项ID | 文档ID | 文档ID |  |  |  |  |  |
| 1 | 1;2;1:10 |  |  |  |  |  |  |

注：

**文档ID**格式：

<DocID>;<TF>;<POS1>:<POS2>

其中**位置信息**取词项在正向索引中的index号

倒排索引存于数据库中，启动服务时再读入内存。

**数据处理流程**

Spider部分：

1. 爬取原始html网页
2. 进行过滤**（**去除<script>等无用信息**）**
3. 存储为**原始网页库**（采用法2存储）

Indexer部分：

1. 读取**原始网页库**
2. 先抽取关键信息（<tittle>等信息），生成**网页信息表**
3. 再将html标签清除，提取正文（全部内容））
4. 对正文进行分词
5. 生成**正向索引**（同时生成词项ID-词项映射表）
6. 生成**倒排索引**（合并相同词项，统计TF，POS）
7. 网页信息表、正向索引、倒排索引，均需存储到数据库

Query部分：

1. （服务器启动时读入**倒排索引**）
2. 对查询词进行分词得到**词项ID集合**
3. 根据词项ID集合检索**倒排索引**，合并之后得到**文档ID集合**(包含TF)
4. 根据TF/POS等对**文档ID集合**进行排序
5. 根据**文档ID集合**从**网页信息表**中获取相关字段，显示于结果页面。
6. 根据网页信息表中的偏移等，从**原始网页库**读取相应网页实现快照功能

**Tips**

1. 关于倒排索引的构建：

上述Indexer流程概括了倒排索引的构建，实际上可分为tittle倒排索引，摘要倒排索引，正文倒排索引。便于之后Query部分的排序。

1. 原始网页库存为少数几个文件在进行结果显示的时候效率会比较高，不必打开众多文件。
2. 每个数据结构都的存储要进行数据库操作，实际是读写磁盘处理，会降低索引构建速度。可考虑若数据集不大，可在query前先读入内存。