并行计算大作业的零碎笔记

2014年5月27日 17:51

我们要**记录**下来自己在做大作业过程中遇到并解决的问题,这样在讲的时候才有东西可讲.才能让老师知道我们做了什么---- 哪些是别人已经做好的,哪些是我们自己搞的.

1) One of the great things about Nutch2.x is that its database is more easily assessible than it was under Nutch 1.x. In particular, if you are are using Nutch 2.0 with MySQL you can see what is going on in the background. This makes both learning and debugging Nutch 2.0 significantly easier than previous versions.
2)草, robots.txt里面写着不能爬公文通! 我们需要看看怎么让nutch不遵守robots.txt
3)conf/regex-urlfilter.txt文件里面有些过滤规则不适合我们,文件自己也说是为了适合爬整个因特网而写的, 比如忽略有3个或以上斜杠/的url. 我们需要看看这里面的规则,并制定合适自己的规则.

Nutch中MapReduce的分析

这个文章讲了如何在nutch中使用MapReduce.

来自 <http://blog.csdn.net/jiutao_tang/article/details/6531254>

Nutch各种参数详解

Nutch基本命令解释（一） （二） （三）
Nutch的命令详解

Lucene / Solr 开发经验

这个文章介绍了Solr的工作原理和内部结构和对应的Java代码、类名，是详细和有价值的文章。

来自 <http://clayz.iteye.com/blog/240357>

Nutch配置文件nutch-default.xml中有用的参数:
(不要直接改这个文件,而是在nutch-site.xml里面覆盖这些配置)
db.ignore.external.links
file.content.limit
http.content.limit
fetcher.server.delay(它就是我们看到的等5秒,太长,要改.)
fetcher.threads.per.queue
fetcher.threads.fetch(可以被命令中的参数覆盖)

How can I recover an aborted fetch process?

... 见原链接.
来自 <http://wiki.apache.org/nutch/FAQ>

I have a big fetchlist in my segments folder. How can I fetch only some sites at a time?

来自 <http://wiki.apache.org/nutch/FAQ>

搜索"Solr,Nutch二次开发"

1. 增量式抓取是指在具有一定量规模的网络页面集合的基础上，采用更新数据的方式选取已有集合中的过时网页进行抓取，以保证所抓取到的数据与真实网络数据足够接近。所以，**我们应该对常常更新的网页如公文通进行增量式爬取， 对别的网页进行累积式抓取**（即抓取机器所能存储和处理的所有网页）
怎么对某一些网页(如公文通)进行增量式抓取呢?
见文章 how-to-re-crawl-with-nutch (但是这个文章没说要改配置文件,实际上是要改的.需要把db.fetch.schedule.class属性值改成**org.apache.nutch.crawl.AdaptiveFetchSchedule**)

现在我们只解决了nutch增量抓取的问题,那么Solr也需要增量索引, 不然每次重建索引太费时了, 要怎么办呢?
* 我觉得不需要担心这个问题. 因为我从一个脚本推断出Solr本身就是增量索引的: 在nutch自己提供的爬虫脚本bin/crawl中, 召唤solr去索引的命令是放在主循环中的, 而且该命令的参数指定的是这个循环新生成的segment. 显然一个新segment只是全体数据的一部分. 所以可以看出solr是增量索引的. 另外召唤solr让它给索引去重的命令(clean命令---remove HTTP 301 and 404 documents and duplicates from indexing backends configured via plugins)也是在主循环中,更可以看出solr是增量索引的, 增加新索引信息后需要给索引去重嘛.

(已经懂,只剩粗体部分有价值)"topN" 参数:
It doesn't mean the branching factor at each depth(and in fact, each "depth" refers to each "segment). If so, segments will grow exponentially at each depth. "topN" means, if a scoring plugin is used, it crawls urls that have topN score, and if there is no score it's simply sorted alphabetically.

So the crawling algorithm is not DFS or BFS, and -depth 5 -topN 100 simply means Nutch will crawl 100 webpages and repeat it 5 times.

这个帖子很好地解答了这个问题:<http://lucene.472066.n3.nabble.com/The-quot-topN-quot-parameter-in-nutch-crawl-td4023321.html>

这个帖子里面的更多信息:

- **With some tuning to a scoring filter you can do whatever you want but in the end everything is going to be crawled (if there are enough resources).**

"Nutch generates fetch lists from the CrawlDB which is nothing more than a sorted list of URL (score, than alphabetically). It just picks the first eligible URL in the sorted list. You really should take a good look at the Generator code, it'll answer most of your questions."

"so the best way to think about "-depth 5 -topN 100" would be "5 batches of 100 ULRs", correct?"

"Yes :) "

"adddays" 参数: 如果指定了这个参数,就会让nutch以为当前时间是(真正的当前时间+adddays天). Fetch Time在fetch时被更新为下次要抓取的时间, 也就是定义一个网页已经太旧要重新抓取的时间. nutch检查哪些urls的fetch time超过当前时间, 需要重新抓取。所以-adddays如果是负数, 将缩小抓取范围。

来自 <http://leibnitz.iteye.com/blog/1130975>