

基于 Heritrix 的面向特定主题的聚焦爬虫研究

朱 敏, 罗省贤

(成都理工大学 信息科学与技术学院, 四川 成都 610059)

摘 要:通过分析 Heritrix 开源爬虫的组件结构, 针对 Heritrix 开源爬虫项目存在的问题, 项目设计了特定的抓取逻辑和定向抓取包含某一特定内容的网页的类, 并引入 BKDRHash 算法进行 URL 散列, 实现了面向特定主题的网页信息搜索, 达到了提高搜索数据的效率以及多线程抓取网页的目的。最后对某一特定主题的网页进行分析, 并进行网页内容抓取, 采用 HTMLParser 工具将抓取的网页数据源转换成特定的格式, 可为面向主题的搜索信息系统以及数据挖掘提供数据源, 为下一步研究工作做好准备。

关键词:聚焦爬虫; Heritrix; BKDRHash 算法; HTMLParser; 搜索引擎

中图分类号: TP31

文献标识码: A

文章编号: 1673-629X(2012)02-0065-04

Research of a Focused Crawler to Specific Topic Based on Heritrix

ZHU Min, LUO Sheng-xian

(School of Information Science and Technology, Chengdu University of Technology, Chengdu 610059, China)

Abstract: By analyzing the Heritrix open-source crawler's component architecture, on account of the existed problems of the Heritrix open-source project, the project designs specific capture logics and classes that can directly crawl particular content pages, implements search for particular topic pages; And introduce the BKDRHash algorithms to URL hashing to achieve a particular topic pages for information search and improve the efficiency of the search data, and achieve the purpose of multi-threaded web crawler. Finally, analyse a particular topic pages and capture content, use HTMLParser tool to crawl the web data source into a specific format, the search can provide a data source for the topic-oriented information systems and data mining, prepare a good potential for further research.

Key words: focused crawler; Heritrix; BKDRHash algorithm; HTMLParser; search engine

0 引言

随着互联网的迅速发展以及 WEB2.0 的普及, 网络信息的膨胀速度呈指数级增长, 各类网站都需要增强检索功能, 通用的搜索引擎已经不能满足不同用户对不同信息的需求。例如在电子商务领域, 如何快速、准确地搜索用户需要的信息, 成为电子商务网站友好性评价的一个关键因素, 因此设计一个针对电子商务网站的个性化搜索引擎^[1,2]显得非常重要。搜索引擎是从各种网络资源中根据关键字搜索出特定的资源的一种工具^[3], 而网络爬虫技术是搜索引擎中的关键技术^[4], 基于 Heritrix 的面向特定主题的聚焦爬虫的研究正是围绕这一应用领域所产生的。

由于 Heritrix^[5] 网络爬虫会将所爬过的网页的所

有信息都抓取到本地硬盘上, 其中大部分内容并非有意义的信息, 文中在 Heritrix 的基础上, 扩展了 FrontierScheduler 实现特定网页的抓取, 并针对 Heritrix 抓取速度缓慢的问题引入了 BKDRHash 散列算法来提高抓取效率。

1 传统网络爬虫与聚焦爬虫

网络爬虫 (Spider) 是一种按照一定的规则, 自动抓取万维网信息的程序或脚本^[6]。Spider 从一个或若干初始网页的 URL 开始, 利用 HTTP 等标准协议读取文档, 将文档中所包括的 URL 放入 URL 队列中, 然后从 URL 队列中新的 URL 处开始进行漫游, 把爬过的网页搜集起来, 直到没有满足条件的新的 URL^[7] 为止。它的目标是尽可能大范围地搜索网络信息, 因此不能满足用户对某一领域或相关主题快速查询的需求。

聚焦爬虫 (Focused Crawler)^[8,9] 需要根据一定的网页分析算法过滤与主题无关的链接, 保留有用的链接并将其放入等待抓取的 URL 队列。然后它将根据一定的搜索策略从队列中选择下一步要抓取的网页

收稿日期: 2011-06-08; 修回日期: 2011-09-17

作者简介: 朱 敏 (1986-), 女, 湖南长沙人, 硕士, 主要研究方向为高性能计算领域中的网络并行计算、搜索引擎中网络爬虫研究; 罗省贤, 教授, 主要研究方向是高性能计算领域中的网络并行计算、网格计算, 信号及信息处理领域中的数字信号处理方法研究及软件开发、信号及图像非线性处理。

URL,并重复上述过程,直到达到系统的某一条件。

2 Heritrix 的架构及工作组件

2.1 Heritrix 简介

Heritrix 是一个专门为互联网上的网页进行存档而开发的网页检索器,是一个完全开源的、使用 JAVA 编写的可扩展的 Web 爬虫项目,开发者利用其出色的可扩展性可以扩展它的各个组件来实现自己的抓取逻辑^[10]。其工作流程为:从 URI 队列中选择一个 URI,根据选定的 URI 下载远程文件,然后分析、归档下载到的内容并写入磁盘镜像目录,再根据一定的策略从分析后的内容中选择 URI 加入 URI 队列,然后不断进行上述工作^[11]。Heritrix 的构架如图 1 所示^[12],其中处理链(Processor chains)由几个处理器组成,如图 2 所示。

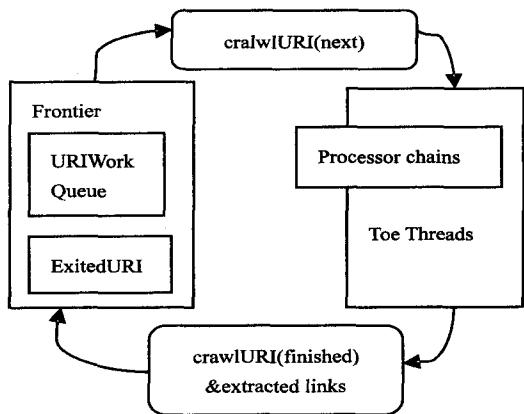


图 1 Heritrix 构架图

2.2 Heritrix 主要组件包含的类

Heritrix 的主要组件为:抓取任务 CrawlOrder、中央控制器 CrawlController、Frontier 链接制造工厂、处理链和 Processor。各组件的主要功能及所包含的类和方法如表 1 所示。

表 1 Heritrix 主要组件

组件	组件的作用	所在的包和相关的类	相关参数和调用的方法
CrawlOrder	读取 xml. order 文件,决定抓取工作的起点	org. archive. crawler. settings XMLSettingHandler	order. xml 文件 XMLSettingHandle() getOrder()
CrawlController	核心组件包 CrawlOrder,包括组件: CrawlScope, ProcessorChinList, Frontier, TeoPool, ServerCache 决定抓取任务的起点和结束	org. archive. crawler. framwok CrawlController	参数: order. xml initialize(SettingHandler) 方法 requestCrawlStart()
Frontier	向线程提供链接	BdbFrontier BdbMultipleWorkQueue BdbWorkQueue BdbUriUnipFilde	next() finished()
Processing chains	下载网页,提取 URI	PreProcessor; Fetch; Extractor Writer PostProcessor	FrontierScheduler()

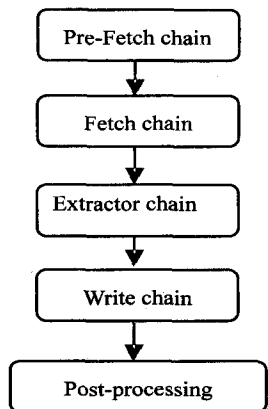


图 2 Processor chains 处理链

3 面向特定主题的聚焦爬虫的设计

3.1 定制 Queue-assignment-policy

Heritrix 使用 Berkely DB 构建链接队列,这些队列放置在 BdbMultipleWorkQueues 中时是以 Key - Value 对存在的,先赋予一个 Key,相同 Key 值的链接放在一起,成为一个 queue。在默认情况下的 Heritrix 使用 HostnameAssignmentPolicy 根据域名(Hostname)产生 key 值,因此相同 Hostname 下的 URL 放在同一个队列中。而文中的实验抓取对象为电子商务网站,这类网站的特点是大部分的 URL 来自于同一个域名,这就造成了某一个队列太长的情况。当一个线程从该队列中获取一个 URL 链接后,这个队列就会处于阻塞状态,直到该链接处理完才从队列的头部取出下一个链接,而线程池中的其他线程因为没有可取的 URL 而处于等待状态。装有大部分 URL 链接的队列会在很长的时间内处于阻塞状态,从而造成 Heritrix 的抓取效率低下,会在某个时间段过后处于一直没有进度的状态。因此,需要改变 Key 值的生成方式,使得所有的 URL

比较平均地散列到不同的队列中,以提高抓取效率。

文中在 Heritrix 中扩展 queue-assignment-policy,实现一个继承自 QueueAssignmentPolicy 的类,覆写其中的 getClassKey() 方法。该方法将一个链接对象处理后,再调用散列算法生成一个 Key 值,相同 Key 的链接存于同一个队列中。散列算法有多种,范先爽在文献“基于 Heritrix 网络爬虫算法的研究与应用”中引入 ELFHash 算法进行 URL 散列。文中采用散列程度高且易使用的 BKDRHash 算法生成 Key 值。覆写的 getClassKey() 的主要部分代码如下:

```
//覆写的 getClassKey()
Public Class getClassKey() (CrawlCon-
troller controller,CandidateURI cauri) {
    String uri = cauri.getUURI().toString
();
    long hash = uri.BKDRHash(uri); //
利用 BKDRHash 算法为 URI 分配 key 值
    String a = Long.toString(hash%50);
    //50 个线程,对应 50 个不同的 URI 处理队列
    return a;
}

//BKDRHash 算法
public long BKDRHash(String str)
{
    long seed = 131; // 31 131 1313 13131 131313 etc..
    long hash = 0;
    for(int i = 0; i < str.length(); i++)
    {
        hash = (hash * seed) + str.charAt(i);
    }
    return hash;
}
```

实践证明,引入 BKDRHash 算法后,抓取效率提高了很多。

3.2 北斗手机页面分析

文中以手机销售网站北斗手机网(www.139shop.com)作为研究对象,进行页面信息分析。在 139shop 网站里,手机是按照品牌进行分类的,分析出北斗手机网手机品牌汇集页面地址为: http://mobile.139shop.com/brand/。查看此页面源码,得到与单个手机品牌页面相关的 URI 地址内容:联想,由于此地址不是完整的 URI 地址,因此使用 Heritrix 将抓取不到单个手机品牌页面的内容。通过从浏览器的地址栏上可以得知,联想手机实际页面为: http://mobile.139shop.com/brand/72/,如果从此页面中单击某个手机,就转到单

个手机的页面: http://mobile.139shop.com/mobile/72/18366.htm,此页面下的信息才是需要保存的。所以在此需要扩展 Heritrix,开发出自己的类,分析出单个手机页面的 URL,并保持此类页面的信息,包括:手机类型、外观设计、上市日期、手机制式、支持频段、铃声系统等数据,图 3 为北斗手机网手机大全列表,图 4 为联想 3GW101 详细信息。

国产手机大全					共收录了 284 个国产手机品牌
联想	CCPO 酷派	多普达	HTC	天语	
海信	酷派	金立	华为	长虹	
步步高	中兴	koobee	ZORA	OPPO	
海尔	知心	TCL	波导	七喜	
魅族	sins	纽曼	奥克斯	康佳	
ADSL	创维	万利达	迪士尼	朵唯	
GT 佳通	国信通	魅族	ORVAP	金鹏	
琦基	海信	贝尔本	爱国者	大显	
柒立特	高斯奇	聆韵	新中桥	亿通	
汇讯	盛乾	夏新	OBEE	亿城	
迪泰元	亿和通	高科	众一	知己迅联	
三新	ZTC 中天	耀利通	三巨网	CECT	
国虹	赛讯达	唯科	兆讯达	友利通	

图 3 北斗手机网手机大全列表

上市日期:	2010年9月
手机类型:	经济
外观设计:	直板
屏幕参数:	26万色 TFT屏;480×800pix;
网络模式:	GSM, 联通 3G (WCDMA)
数据业务:	/3G//GPRS
支持频段:	WCDMA /850/900/1800/1900MHz
操作系统:	系统类型:Android;
存储卡:	T-Flash/MicroSD卡;
输入方式:	支持手写输入
机身颜色:	珍珠红、黑色
产品尺寸:	120mm×61mm×12.5mm
标准配置:	锂电池;充电器

图 4 手机详细页面信息

3.3 开发特定的抓取类 Extractor 和扩展 Scheduler

在 HeritrixProject 项目下建立 my.extractor 包,在此包内新建类 Mobile139Extractor,该类继承父类 Extractor,覆写 extract 方法。在 extract(CrawlURI curi)方法中判断传入其中的参数是否为北斗手机网的所有手机品牌汇集页面,如果是,则解析出页面下的链接部分的/brand/后的 id/,并在 id 号前加上 http://mobile.139shop.com/brand/,生成对应 id 号的手机品牌汇集页面地址: http://mobile.139shop.com/brand/id/,然后将其加到等待 FrontierScheduler 处理的列表中,以待处理。

扩展 FrontierScheduler 实现特定网页的抓取,在 HeritrixProject 项目下建立 my.processor 包,在此包下新建类 FrontierSchedulerForMobile139,该类继承父类 FrontierScheduler,重写 schedule() 方法,只有满足条件

的 URL 才允许加入到等待队列中。扩展新类后, Heritrix 中扩展的类及主要方法如表 2 所示。

表 2 Heritrix 中扩展类及主要方法说明

所属的包	父类	新类	主要方法说明
my. extractor	Extractor	Moble	extract(CrawlURI curi), 用来过滤传入的 URI 地址以及生成某个品牌页面的 URI
		139Extractor	line. substring(line. indexOf("brand")+6, line. indexOf("target")-3)+"/"
my. processor	FrontierScheduler	FrontierSchedulerForMoble	schedule(CandidateURI caUri), 将满足条件的 URI 加入等待队列
		139	getController(). getFrontier(). schedule(caUri)

在 modules 文件夹中的 Processor. options 模板文件中添加 Moble139Extractor 和 FrontierSchedulerForMoble139 后, 再在 Modules 页面中心选择 my. extractor. Moble139Extractor 和 my. processor. FrontierSchedulerForMoble139, 就可在 Heritrix 中实现特定逻辑定制。

4 应用实例

文中采用上述方法, 设计与实现了基于 Heritrix 的面向特定主题的聚焦爬虫, 抓取网页后建立一个镜像目录存放某一手机网页的信息。

由于 HTML 中的标签不一定成对出现, Web 页面中主要部分的格式编排不合理, 虽然浏览器也能适应这种不完美且复杂的格式, 正确地显示其中的内容, 但对于用户则很难从中提取数据。文中引入一个开源项目 HTMLParser 来解析网页, 将所有手机网页下的原始 URL 地址、手机类型、外观设计等描述, 以及产品的图片提取出来, 图片文件名是经 Hash 算法转换后的字符串, 所有图片存在同一个目录下, 供用户查询相关产品时显示。它与抓取目标的描述、抓取目标的分析、用户的查询方式相联系, 尽快地发现用户感兴趣的资源, 提高 Web 信息挖掘的效率^[7,13]。

5 结束语

目前, 搜索引擎技术越来越受关注, 其应用领域也越来越广。文中所设计的面向北斗手机网的聚焦网络爬虫的扩展应用, 可针对某一特定主题快速搜集数据, 并且该方法具有通用性, 易于移植到其他电子商务网站上应用, 可为电子商务的数据挖掘准备数据源。

参考文献:

- [1] 严莉莉, 王倩倩, 孟 杰, 等. 基于聚类的个性化元搜索引擎设计[J]. 计算机技术与发展, 2007, 17(4): 186-188.
- [2] 王 萍, 刘 军, 姚笑秋. 基于小型搜索引擎的个性化策略研究[J]. 计算机技术与发展, 2007, 17(11): 36-38.
- [3] 沈贺丹, 潘亚楠, 邵良杉. 关于搜索引擎的研究综述[J]. 计算机技术与发展, 2006, 16(4): 147-149.
- [4] Pinkerton B. Finding what people want: experiences with the web crawler[C]//Proceedings of the Second World-Wide Web Conference. Chicago, Illinois: [s. n.], 1994.
- [5] Heritrix 官方网站[EB/OL]. [2011-04]. <http://crawler.archive.org>.
- [6] Guo Q, Guo H, Zhang Z Q, et al. Schema Driven Topic Specific Web Crawling[C]//DASFAA. [s. l.]: [s. n.], 2005.
- [7] 周立柱, 林 玲. 聚焦爬虫技术研究综述[J]. 计算机应用, 2005, 25(9): 1965-1969.
- [8] 唐 苏, 刘 循. 基于超链接引导和链接图分析的主题搜索引擎[J]. 计算机技术与发展, 2011, 21(2): 155-158.
- [9] Dong H, Hussain F K. Focused Crawling for Automatic Service Discovery, Annotation and Classification in Industrial Digital Ecosystems[J]. IEEE Trans on Industrial Electronics, 2011, 58(6): 2106-2116.
- [10] 李 刚, 宋 伟. 征服 Ajax+Lucene 构建搜索引擎[M]. 北京: 人民邮电出版社, 2006.
- [11] 邱 哲, 符滔滔. Lucene 2.0+Heritrix 开发自己的搜索引擎[M]. 北京: 人民邮电出版社, 2007.
- [12] 杨 颂, 欧阳柳波. 基于 Heritrix 的面向电子商务网站增量爬虫研究[J]. 软件导刊, 2010, 9(7): 38-39.
- [13] 杨定中, 赵 刚, 王 泰. 网络爬虫在 Web 信息搜索与数据挖掘中的应用[J]. 计算机工程与设计, 2009, 30(24): 5658-5662.

(上接第 64 页)

- environments[J]. ITSAP, 1999, 1(7): 55-58.
- [3] Ghitza O. Auditory models and human performance in tasks related to speech coding and speech recognition[J]. IRSAP, 1994, 1(2): 113-131.
- [4] 李霄寒, 戴蓓倩, 方绍武. 高阶 MFCC 的话者识别性能及其噪声鲁棒性[J]. 信号处理, 2001, 17(2): 124-129.
- [5] Shaughnessy D. Speech Communication[M]. Reading, MA: Addison Wesley, 1987: 150-153.
- [6] 郝 静. 基于粒计算的语音实时分段算法[D]. 太原: 太原理工大学, 2008.
- [7] 张 刚, 张雪英, 马建芬. 语音处理与编码[M]. 北京: 兵器

工业出版社, 2000.

- [8] 梁五洲. 抗噪语音识别特征提取算法的研究[D]. 太原: 太原理工大学, 2006.
- [9] 赵 力. 语音信号处理[M]. 北京: 机械信号处理出版社, 2003.
- [10] 刘雅琴, 智爱娟. 几种语音识别特征参数的研究[J]. 计算机技术与发展, 2009, 19(12): 67-70.
- [11] 沈江峰. 8kbit/s 低延迟语音编码算法研究[D]. 太原: 太原理工大学, 2007.
- [12] 杨 海. 感知语音质量评价 PESQ 及其在通信系统中的应用[J]. 江西通信科技, 2004(2): 46-47.