ISSN 1009-3044
Computer Knowledge and Technology 电脑知识与技术
Vol.6,No.15, May 2010, pp.4112-4115

E-mail: kfyj@cccc.net.cn
http://www.dnzs.net.cn
Tel:+86-551-5690963 5690964

# 网络爬虫技术的研究

孙立伟,何国辉,吴礼发

(解放军理工大学 指挥自动化学院,江苏 南京 210007)

**摘要**:网络信息资源的迅猛增长使得传统搜索引擎已经无法满足人们对有用信息获取的要求,作为搜索引擎的基础和重要组成部分,网络爬虫的作用显得尤为重要,该文介绍了网络爬虫的基本概念、爬行 Web 面临的困难及应对措施,其次从体系结构、爬行策略和典型应用等方面研究了通用网络爬虫、聚焦网络爬虫、增量式网络爬虫和深层网络爬虫四种常见网络爬虫,最后指出了进一步工作的发展方向。

**关键词**:搜索引擎;网络爬虫

**中图分类号**:TP393 **文献标识码**:A **文章编号**:1009-3044(2010)15-4112-04

**Research on the Web Crawler**

SUN Li-wei, HE Guo-hui, WU Li-fa

(Institute of Command and Automation of PLAUST, Nanjing 210007, China)

**Abstract:** The traditional search engines can not satisfy the demands of getting useful information with the blast developing of information resources on Internet, as the foundation and important part of search engine, the action of the Web Crawler appears especially important, the article introduces the concept of Web crawler, the trouble of crawling and the resolvent, also have a research on four kinds of familiar Web Crawler.

**Key words:** search engines; web crawler

网络爬虫(Web Crawler),又称为网络蜘蛛(Web Spider)或 Web 信息采集器,是一个自动下载网页的计算机程序或自动化脚本,是搜索引擎的重要组成部分。网络爬虫通常从一个称为种子集的 URL 集合开始运行,它首先将这些 URL 全部放入到一个有序的待爬行队列里,按照一定的顺序从中取出 URL 并下载所指向的页面,分析页面内容,提取新的 URL 并存入待爬行 URL 队列中,如此重复上面的过程,直到 URL 队列为空或满足某个爬行终止条件,从而遍历 Web[1]。该过程称为网络爬行(Web Crawling)。

## 1 网络爬虫面临的问题

截止到 2007 年底,Internet 上网页数量超出 160 亿个,研究表明接近 30%的页面是重复的;动态页面的存在:客户端、服务器端脚本语言的应用使得指向相同 Web 信息的 URL 数量呈指数级增长。上述特征使得网络爬虫面临一定的困难,主要体现在 Web 信息的巨大容量使得爬虫在给定时间内只能下载少量网页。Lawrence 和 Giles 的研究[2]表明没有哪个搜索引擎能够索引超出 16%的 Internet 上 Web 页面,即使能够提取全部页面,也没有足够的空间来存储。

为提高爬行效率,爬虫需要在单位时间内尽可能多的获取高质量页面,是它面临的难题之一。当前有五种表示页面质量高低的方式[1]:Similarity(页面与爬行主题之间的相似度)、Backlink(页面在 Web 图中的入度大小)、PageRank(指向它的所有页面平均权值之和)、Forwardlink(页面在 Web 图中的出度大小)、Location(页面的信息位置);Parallel(并行性问题)[3]。为了提高爬行速度,网络通常会采取并行爬行的工作方式,随之引入了新的问题:重复性(并行运行的爬虫或爬行线程同时运行时增加了重复页面)、质量问题(并行运行时,每个爬虫或爬行线程只能获取部分页面,导致页面质量下降)、通信带宽代价(并行运行时,各个爬虫或爬行线程之间不可避免要进行一些通信)。并行运行时,网络爬虫通常采用三种方式:独立方式(各个爬虫独立爬行页面,互不通信)、动态分配方式(由一个中央协调器动态协调分配 URL 给各个爬虫)、静态分配方式(URL 事先划分给各个爬虫)。

## 2 网络爬虫的分类

网络爬虫按照系统结构和实现技术,大致可以分为以下几种类型:通用网络爬虫(General Purpose Web Crawler)、聚焦网络爬虫(Focused Web Crawler)、增量式网络爬虫(Incremental Web Crawler)、深层网络爬虫(Deep Web Crawler)。实际的网络爬虫系统通常是几种爬虫技术相结合实现的。

### 2.1 通用网络爬虫

通用网络爬虫[3]又称全网爬虫(Scalable Web Crawler),爬行对象从一些种子 URL 扩充到整个 Web,主要为门户站点搜索引擎和大型 Web 服务提供商采集数据。由于商业原因,它们的技术细节很少公布出来。这类网络爬虫的爬行范围和数量巨大,对于爬行速度和存储空间要求较高,对于爬行页面的顺序要求相对较低,同时由于待刷新的页面太多,通常采用并行工作方式,但需要较长时

收稿日期:2010-03-11

间才能刷新一次页面。虽然存在一定缺陷,通用网络爬虫适用于为搜索引擎搜索广泛的主题,有较强的应用价值。

通用网络爬虫的结构大致可以分为页面爬行模块、页面分析模块、链接过滤模块、页面数据库、URL 队列、初始 URL 集合几个部分,其体系结构如图 1 所示[4]。

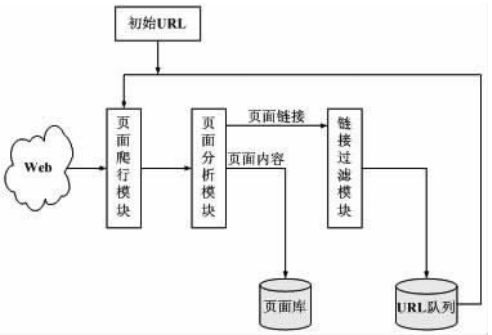

图 1 通用网络爬虫体系结构

为提高工作效率,通用网络爬虫会采取一定的爬行策略。常用的爬行策略有[5]:深度优先策略、广度优先策略。

1) 深度优先策略:其基本方法是按照深度由低到高的顺序,依次访问下一级网页链接,直到不能再深入为止。爬虫在完成一个爬行分支后返回到上一链接节点进一步搜索其它链接。当所有链接遍历完后,爬行任务结束。这种策略比较适合垂直搜索或站内搜索,但爬行页面内容层次较深的站点时会造成资源的巨大浪费;

2) 广度优先策略:此策略按照网页内容目录层次深浅来爬行页面,处于较浅目录层次的页面首先被爬行。当同一层次中的页面爬行完毕后,爬虫再深入下一层继续爬行。这种策略能够有效控制页面的爬行深度,避免遇到一个无穷深层分支时无法结束爬行的问题,实现方便,无需存储大量中间节点,不足之处在于需较长时间才能爬行到目录层次较深的页面。

典型的通用爬虫有 Google Crawler、Mercator。Google Crawler[6]是一个分布式的基于整个 Web 的爬虫,采用异步 I/O 而不是多线程来实现并行化。它有一个专门的 URL Server 进程负责为多个爬虫节点维护 URL 队列。Google Crawler 还使用了许多算法优化系统性能,最著名的就是 PageRank 算法。

### 2.2 聚焦网络爬虫

聚焦网络爬虫(Focused Crawler),又称主题网络爬虫(Topical Crawler),是指选择性地爬行那些与预先定义好的主题相关页面的网络爬虫[8]。和通用网络爬虫相比,聚焦爬虫只需要爬行与主题相关的页面,极大地节省了硬件和网络资源,保存的页面也由于数量少而更新快,还可以很好地满足一些特定人群对特定领域信息的需求[3]。

聚焦网络爬虫和通用网络爬虫相比,增加了链接评价模块以及内容评价模块,其系统结构图 2 所示[9]。

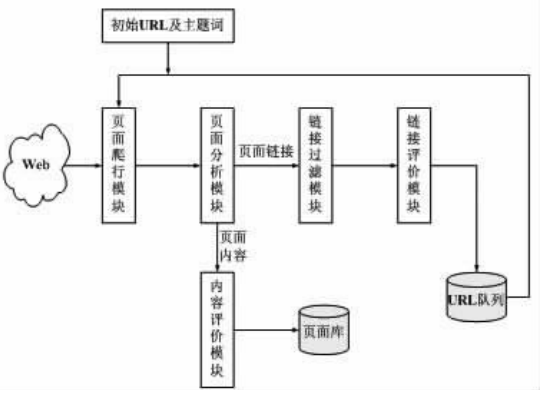

图 2 聚焦网络爬虫体系结构

聚焦爬虫爬行策略实现的关键是评价页面内容和链接的重要性,不同的方法计算出的重要性不同,由此导致链接的访问顺序也不同。

1) 基于内容评价的爬行策略:DeBra[10]将文本相似度的计算方法引入到网络爬虫中,提出了 Fish Search 算法,它将用户输入的查询词作为主题,包含查询词的页面被视为与主题相关,其局限性在于无法评价页面与主题相关度的高低。Herseovic[11]对 Fish Search 算法进行了改进,提出了 Shark-search 算法,利用空间向量模型计算页面与主题的相关度大小;

2) 基于链接结构评价的爬行策略:Web 页面作为一种半结构化文档,包含很多结构信息,可用来评价链接重要性。PageRank 算法最初用于搜索引擎信息检索中对查询结果进行排序,也可用于评价链接重要性[12],具体做法就是每次选择 PageRank 值较大页面中的链接来访问。另一个利用 Web 结构评价链接价值的方法是 HITS 方法[13],它通过计算每个已访问页面的 Authority 权重和 Hub 权重,并以此决定链接的访问顺序。

3) 基于增强学习的爬行策略[14]:Rennie 和 McCallum 将增强学习引入聚焦爬虫,利用贝叶斯分类器,根据整个网页文本和链接文本对超链接进行分类,为每个链接计算出重要性,从而决定链接的访问顺序;

4) 基于语境图的爬行策略:Diligenti 等人提出了一种通过建立语境图[15](Context Graphs)学习网页之间的相关度,训练一个机器学习系统,通过该系统可计算当前页面到相关 Web 页面的距离,距离越近的页面中的链接优先访问。

印度理工大学(IIT)和 IBM 研究中心的研究人员开发了一个典型的聚焦网络爬虫[8]。该爬虫对主题的定义既不是采用关键词也不是加权矢量,而是一组具有相同主题的网页。它包含两个重要模块:一个是分类器,用来计算所爬行的页面与主题的相关度,确定是否与主题相关;另一个是净化器,用来识别通过较少链接连接到大量相关页面的中心页面。

### 2.3 增量式网络爬虫

增量式网络爬虫(Incremental Web Crawler)[16]是指对已下载网页采取增量式更新和只爬行新产生的或者已经发生变化网页的爬虫,它能够在一定程度上保证所爬行的页面是尽可能新的页面。和周期性爬行和刷新页面的网络爬虫相比,增量式爬虫只会在需要的时候爬行新产生或发生更新的页面,并不重新下载没有发生变化的页面,可有效减少数据下载量,及时更新已爬行的网页,减小时间和空间上的耗费,但是增加了爬行算法的复杂度和实现难度。

增量式网络爬虫的体系结构[16]如图 3 所示,它包含爬行模块、排序模块、更新模块、本地页面集、待爬行 URL 集以及本地页面URL 集。

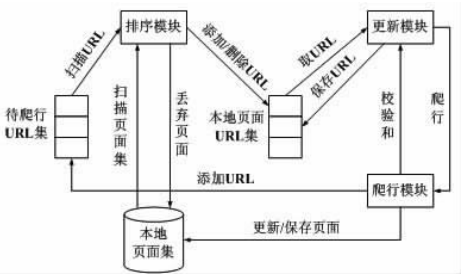

图 3 增量式爬虫体系结构

增量式爬虫有两个目标:保持本地页面集中存储的页面为最新页面和提高本地页面集中页面的质量。为实现第一个目标,增量式爬虫需要通过重新访问网页来更新本地页面集中页面内容,常用的方法有:

1) 统一更新法[17]:爬虫以相同的频率访问所有网页,不考虑网页的改变频率;

2) 个体更新法[17]:爬虫根据个体网页的改变频率来重新访问各页面;

3) 基于分类的更新法[18]:爬虫根据网页改变频率将其分为更新较快网页子集和更新较慢网页子集两类,然后以不同的频率访问这两类网页。

为实现第二个目标,增量式爬虫需要对网页的重要性排序,常用的策略有:广度优先策略[19]、PageRank 优先策略等。

IBM 开发的 WebFountain[20]是一个功能强大的增量式网络爬虫,它采用一个优化模型控制爬行过程,并没有对页面变化过程做任何统计假设,而是采用一种自适应的方法根据先前爬行周期里爬行结果和网页实际变化速度对页面更新频率进行调整。

北京大学的天网增量爬行系统[21]旨在爬行国内 Web,将网页分为变化网页和新网页两类,分别采用不同爬行策略。为缓解对大量网页变化历史维护导致的性能瓶颈,它根据网页变化时间局部性规律,在短时期内直接爬行多次变化的网页,为尽快获取新网页,它利用索引型网页跟踪新出现网页。

### 2.4 Deep Web 爬虫

Web 页面按存在方式可以分为表层网页(Surface Web)和深层网页(Deep Web,也称 Invisible Web Pages 或 Hidden Web)。表层网页是指传统搜索引擎可以索引的页面,以超链接可以到达的静态网页为主构成的 Web 页面。Deep Web 是那些大部分内容不能通过静态链接获取的、隐藏在搜索表单后的,只有用户提交一些关键词才能获得的 Web 页面。例如那些用户注册后内容才可见的网页就属于 Deep Web。2000 年 Bright Planet 指出[22]:Deep Web 中可访问信息容量是 Surface Web 的几百倍,是互联网上最大、发展最快的新型信息资源。

Deep Web 爬虫体系结构如图 4 所示,包含六个基本功能模块[24](爬行控制器、解析器、表单分析器、表单处理器、响应分析器、LVS 控制器)和两个爬虫内部数据结构(URL 列表、LVS 表)。其中 LVS(Label Value Set)表示标签/数值集合,用来表示填充表单的数据源。

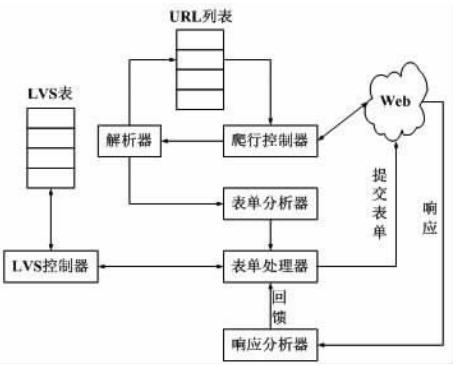


图 4 Deep Web 爬虫体系结构

Deep Web 爬虫爬行过程中最重要部分就是表单填写,包含两种类型[23]:

1) 基于领域知识的表单填写:此方法一般会维持一个本体库,通过语义分析来选取合适的关键词填写表单。Yiyao Lu[25]等人提出一种获取 Form 表单信息的多注解方法,将数据表单按语义分配到各个组中,对每组从多方面注解,结合各种注解结果来预测一个最终的注解标签;郑冬冬[26]等人利用一个预定义的领域本体知识库来识别 Deep Web 页面内容,同时利用一些来自 Web 站点导航模式来识别自动填写表单时所需进行的路径导航;

2) 基于网页结构分析的表单填写:此方法一般无领域知识或仅有有限的领域知识,将网页表单表示成 DOM 树,从中提取表单各字段值。Desouky 等人[27]提出一种 LEHW 方法,该方法将 HTML 网页表示为 DOM 树形式,将表单区分为单属性表单和多属性表单,分别进行处理;孙彬等人提出一种基于 XQuery 的搜索系统[28],它能够模拟表单和特殊页面标记切换,把网页关键字切换信息描述为三元组单元,按照一定规则排除无效表单,将 Web 文档构造成 DOM 树,利用 XQuery 将文字属性映射到表单字段。

Raghavan 等人提出的 HIWE 系统[24]中,爬行管理器负责管理整个爬行过程,分析下载的页面,将包含表单的页面提交表单处理器处理,表单处理器先从页面中提取表单,从预先准备好的数据集中选择数据自动填充并提交表单,由爬行控制器下载相应的结果页面。

## 3 结束语

网络爬虫所爬行的页面被搜索引擎用来建立索引,以便实现快速搜索;网络爬虫还可用于网站维护任务,检查网站链接有效性和验证源码。网络爬虫技术将用来应对互联网络上日益增多的网络资源和信息需求,处理一些新技术开发的网页,爬行一些全新的信息,面临很大的挑战。比如 AJAX 技术开发的页面就不能被传统爬虫所抓取,本文今后将研究 AJAX 技术开发网页的爬行问题。

## 参考文献:

[1] J. Cho. Crawling the web: Discovery and Maintenance of Large-scale Web Data [D]. L.A.: Stanford University, 2001.

[2] S. Lawrence, C. L. Giles. Accessibility of information on the Web [J]. Nature, 1999.

[3] 李盛韬,余智华,程学旗.Web 信息采集研究进展[J].计算机科学,2003.

[4] 蒋科.基于领域概念定制的主题爬虫系统的设计与实现[D].西安:西安电子科技大学,2007.

[5] 王学松. Lucene + Nutch 搜索引擎开发[M].北京:人民邮电出版社,2008.

[6] S. Brin, L. Page. The Anatomy of a Large-Scale Hypertextual Web Search Engine [C].In Proceedings of 7th International World Wide Web Conference, Brisbane, Australia, 1998.

[7] M Burner. Crawling towards Eternity: Building an archive of the World Wide Web [J].World Wide Web Techniques Magazine,1997.

[8] S. Chakrabarti, M. van den Berg and B. Dom. Focused Crawling: A New Approach to Topic-Specific Web Resource Discovery [C]. In Proceedings of the 8th International World Wide Web Conference, Toronto, Canada, 1999.

[9] 刘洁清.网站聚焦爬虫的研究[D].南昌:江西财经大学,2006.

[10] Bra D, P. Houben, Kornatzky M. Information retrieval in distributed hypertexts [C]. In Proceedings of the 4th RIAO Conference, New York, United Staes, 1994.

[11] Hersovici M, Heydon A, Mitzenmacher M et al. The shark-search algorithm-an application: Tailored web site mapping [C]. In Proceedings of the 7th International World Wide Web Conference, Brisbane, Australia, 1998.

[12] Menczer F. Complementing search engines with online web mining agents [J]. Decision Support System, 2003.

[13] Kleinberg J. Authoritative sources in a hyperlinked environment [J].Journal of the ACM, 1998.

[14] Rennie J, McCallum A. Using reinforcement learning to spider the web efficiently [C]. In Proceedings of the International Conference on Machine Learning, Slovenia, 1999.

[15] M. Diligenti., F. Coetzee, S. Lawrence, et al. Focused crawling using context graphs [C]. In Proceedings of 26th International Conference on Very Large Database, Cairo, Egypt, 2000.

[16] J. Cho, H. Garcia-Molina. The evolution of the web and implications for an incremental crawler [C]. In Proceedings of the 26th International Conference on Very Large Database, Cairo, Egypt, 2000.

[17] A. Arasu, J. Cho, H. Garcia-Molina, et al. Searching the web [J].ACM Transaction on Internet Technology, 2001.

[18] 文坤梅,卢正鼎. 搜索引擎中基于分类的网页更新方法研究[J].计算机科学,2004.

[19] M. Najork, J. L. Wiener. Breadth-first crawling yields high-quality pages [C].In Proceedings of the 10th International Conference on World Wide Web, 2001.

[20] J. Edwards, K McCurley, J Tomlin. An adaptive model for optimizing performance of an incremental web crawler [C].In Proceedings of the 10th International Conference on World Wide Web, 2001.

[21] Yan HF, Wang JY, Li XM, et al. Architectual design and evaluation of an efficient Web-crawling system [J].Journal of Systems and Software, 2002.

[22] M K. Bergman. The Deep Web: Surfaceing Hidden Value [EB/OL]. http://www.completeplanet.com/Tutorials/DeepWeb, 2000.

[23] 曾伟辉,李淼. 深层网络爬虫研究综述[J].计算机系统应用,2008.

[24] S. Raghavan, M. Garcia. Crawling the Hidden Web [C]. In Proceedings of the 27th International Conference on Very Large Database, 2001.

[25] Yiyao Lu, Hai He, Hongkun Zhao, et al. Annotating Structured Data of the Deep Web [C], In IEEE 23rd International Conference on Data Engineering, 2007.

[26] 郑冬冬,赵朋朋,崔志明. Deep Web 爬虫研究与设计[J].清华大学学报:自然科学版,2005.

[27] A. Desouky, A. Hesham. An Automatic Label Extraction Technique for Domain-Specific Hidden Web Crawling [C].In the 2006 International Conference on Computer Engineering and Systems, 2006.

[28] 孙彬,王东,李娟.基于 XQuery 的 Deep Web 搜索系统的设计与实现[J].科学技术与工程,2007.