Web Crawling
Web Crawling

Christopher Olston
Yahoo! Research
701 First Avenue
Sunnyvale, CA, 94089
USA
olston@yahoo-inc.com

Marc Najork
Microsoft Research
1065 La Avenida
Mountain View, CA, 94043
USA
najork@microsoft.com

Full text available at: http://dx.doi.org/10.1561/1500000017

now
the essence of knowledge
Boston – Delft
Foundations and Trends® in Information Retrieval
Volume 4 Issue 3, 2010
Editorial Board

Editor-in-Chief:
Jamie Callan
Carnegie Mellon University
callan@cmu.edu

Fabrizio Sebastiani
Consiglio Nazionale delle Ricerche
fabrizio.sebastiani@isti.cnr.it

Editors
Alan Smeaton (Dublin City University)
Andrei Z. Broder (Yahoo! Research)
Bruce Croft (University of Massachusetts, Amherst)
Charles L.A. Clarke (University of Waterloo)
Ellen Voorhees (National Institute of Standards and Technology)
Ian Ruthven (University of Strathclyde, Glasgow)
James Allan (University of Massachusetts, Amherst)
Justin Zobel (RMIT University, Melbourne)
Maarten de Rijke (University of Amsterdam)
Marcello Federico (ITC-irst)
Norbert Fuhr (University of Duisburg-Essen)
Soumen Chakrabarti (Indian Institute of Technology)
Susan Dumais (Microsoft Research)
Wei-Ying Ma (Microsoft Research Asia)
William W. Cohen (CMU)
Editorial Scope

Foundations and Trends® in Information Retrieval will publish survey and tutorial articles in the following topics:

- Applications of IR
- Architectures for IR
- Collaborative filtering and recommender systems
- Cross-lingual and multilingual IR
- Distributed IR and federated search
- Evaluation issues and test collections for IR
- Formal models and language models for IR
- IR on mobile platforms
- Indexing and retrieval of structured documents
- Information categorization and clustering
- Information extraction
- Information filtering and routing
- Metasearch, rank aggregation and data fusion
- Natural language processing for IR
- Performance issues for IR systems, including algorithms, data structures, optimization techniques, and scalability
- Question answering
- Summarization of single documents, multiple documents, and corpora
- Text mining
- Topic detection and tracking
- Usability, interactivity, and visualization issues in IR
- User modelling and user studies for IR
- Web search

Information for Librarians

Foundations and Trends® in Information Retrieval, 2010, Volume 4, 5 issues. ISSN paper version 1554-0669. ISSN online version 1554-0677. Also available as a combined paper and online subscription.
Web Crawling

Christopher Olston\textsuperscript{1} and Marc Najork\textsuperscript{2}

\textsuperscript{1} Yahoo! Research, 701 First Avenue, Sunnyvale, CA, 94089, USA
olston@yahoo-inc.com
\textsuperscript{2} Microsoft Research, 1065 La Avenida, Mountain View, CA, 94043, USA
najork@microsoft.com

Abstract

This is a survey of the science and practice of web crawling. While at first glance web crawling may appear to be merely an application of breadth-first-search, the truth is that there are many challenges ranging from systems concerns such as managing very large data structures to theoretical questions such as how often to revisit evolving content sources. This survey outlines the fundamental challenges and describes the state-of-the-art models and solutions. It also highlights avenues for future work.
6  Avoiding Problematic and Undesirable Content  51
6.1  Redundant Content  51
6.2  Crawler Traps  52
6.3  Web Spam  53
6.4  Cloaked Content  54

7  Deep Web Crawling  57
7.1  Types of Deep Web Sites  57
7.2  Problem Overview  59
7.3  Content Extraction  59

8  Future Directions  63

References  67
1

Introduction

A web crawler (also known as a robot or a spider) is a system for the bulk downloading of web pages. Web crawlers are used for a variety of purposes. Most prominently, they are one of the main components of web search engines, systems that assemble a corpus of web pages, index them, and allow users to issue queries against the index and find the web pages that match the queries. A related use is web archiving (a service provided by e.g., the Internet archive [77]), where large sets of web pages are periodically collected and archived for posterity. A third use is web data mining, where web pages are analyzed for statistical properties, or where data analytics is performed on them (an example would be Attributor [7], a company that monitors the web for copyright and trademark infringements). Finally, web monitoring services allow their clients to submit standing queries, or triggers, and they continuously crawl the web and notify clients of pages that match those queries (an example would be GigaAlert [64]).

The raison d’être for web crawlers lies in the fact that the web is not a centrally managed repository of information, but rather consists
of hundreds of millions of independent web content providers, each one providing their own services, and many competing with one another. In other words, the web can be viewed as a federated information repository, held together by a set of agreed-upon protocols and data formats, such as the Transmission Control Protocol (TCP), the Domain Name Service (DNS), the Hypertext Transfer Protocol (HTTP), the Hypertext Markup Language (HTML) and the robots exclusion protocol. So, content aggregators (such as search engines or web data miners) have two choices: They can either adopt a pull model where they will proactively scour the web for new or updated information, or they could try to establish a convention and a set of protocols enabling content providers to push content of interest to the aggregators. Indeed, the Harvest system [24], one of the earliest search services, adopted such a push model. However, this approach did not succeed, and virtually all content aggregators adopted the pull approach, with a few provisos to allow content providers to exclude all or part of their content from being crawled (the robots exclusion protocol) and to provide hints about their content, its importance and its rate of change (the Sitemaps protocol [110]).

There are several reasons why the push model did not become the primary means of acquiring content for search engines and other content aggregators: The fact that web servers are highly autonomous means that the barrier of entry to becoming a content provider is quite low, and the fact that the web protocols were at least initially extremely simple lowered the barrier even further — in fact, this simplicity is viewed by many as the reason why the web succeeded where earlier hypertext systems had failed. Adding push protocols would have complicated the set of web protocols and thus raised the barrier of entry for content providers, while the pull model does not require any extra protocols. By the same token, the pull model lowers the barrier of entry for content aggregators as well: Launching a crawler does not require any a priori buy-in from content providers, and indeed there are over 1,500 operating crawlers [17], extending far beyond the systems employed by the big search engines. Finally, the push model requires a trust relationship between content provider and content aggregator, something that is not given on the web at large — indeed, the relationship between
content providers and search engines is characterized by both mutual
dependence and adversarial dynamics (see Section 6).

1.1 Challenges

The basic web crawling algorithm is simple: Given a set of seed Uniform
Resource Locators (URLs), a crawler downloads all the web pages
addressed by the URLs, extracts the hyperlinks contained in the pages,
and iteratively downloads the web pages addressed by these hyperlinks.
Despite the apparent simplicity of this basic algorithm, web crawling
has many inherent challenges:

- **Scale.** The web is very large and continually evolving. Crawlers that seek broad coverage and good freshness must achieve extremely high throughput, which poses many difficult engineering problems. Modern search engine companies employ thousands of computers and dozens of high-speed network links.

- **Content selection tradeoffs.** Even the highest-throughput crawlers do not purport to crawl the whole web, or keep up with all the changes. Instead, crawling is performed selectively and in a carefully controlled order. The goals are to acquire high-value content quickly, ensure eventual coverage of all reasonable content, and bypass low-quality, irrelevant, redundant, and malicious content. The crawler must balance competing objectives such as coverage and freshness, while obeying constraints such as per-site rate limitations. A balance must also be struck between exploration of potentially useful content, and exploitation of content already known to be useful.

- **Social obligations.** Crawlers should be “good citizens” of the web, i.e., not impose too much of a burden on the web sites they crawl. In fact, without the right safety mechanisms a high-throughput crawler can inadvertently carry out a denial-of-service attack.

- **Adversaries.** Some content providers seek to inject useless or misleading content into the corpus assembled by
the crawler. Such behavior is often motivated by financial incentives, for example (mis)directing traffic to commercial web sites.

1.2 Outline

Web crawling is a many-faceted topic, and as with most interesting topics it cannot be split into fully orthogonal subtopics. Bearing that in mind, we structure the survey according to five relatively distinct lines of work that occur in the literature:

- Building an efficient, robust and scalable crawler (Section 2).
- Selecting a traversal order of the web graph, assuming content is well-behaved and is interconnected via HTML hyperlinks (Section 4).
- Scheduling revisitation of previously crawled content (Section 5).
- Avoiding problematic and undesirable content (Section 6).
- Crawling so-called “deep web” content, which must be accessed via HTML forms rather than hyperlinks (Section 7).

Section 3 introduces the theoretical crawl ordering problem studied in Sections 4 and 5 and describes structural and evolutionary properties of the web that influence crawl ordering. Section 8 gives a list of open problems.
References

References


References


References


[76] International Workshop Series on Adversarial Information Retrieval on the Web, 2005–.


References


