International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2015): 6.391

Design Issues in Web Crawlers and Review of Parallel Crawlers

Komal[1], Dr. Ashutosh Dixit[2]

[1]PG Student, Dept. of Computer Engineering, YMCA University of Sc. & Technology, Faridabad (Haryana)

[2]Associate Professor, Dept. of Computer Engineering, YMCA University of Sc. & Technology, Faridabad (Haryana)

Abstract: ***With the increase in size of web, search engine depends upon the Web Crawler to download and build index of million/billion of pages for efficient information retrieval when user interact through search interface. This paper will include the definition of Web Crawler, criteria on the basis of which various types of crawler are defined [4] and some common issues with the design of crawler, parallel crawler, its issues and comparison of solutions provided by researchers.***

Keywords: WWW, URL, Mobile Crawler, Web Crawler, Mobile Agents, Parallel Crawler

1. Introduction

A Crawler is a program which is used for downloading web pages from World Wide Web (which is basically a collection of text documents, images, multimedia and other resources which are linked by HYPERLINKS and URLs) for web search engine (is a software system which is used to search for information on World Wide Web). Web crawler is also called as Spider, Web scutters, Automatic indexer, wanderers, Web robots, ants, bots. As the time is changing size of World Wide Web is also changing, in the recent years it has grown from thousand to billion. Due to this explosion in size, web search engine are becoming increasingly important as they are used for locating information. Basically web search engine depends upon web crawler to create and maintain web indices for web pages. A web crawler works by starting with set of URLs (which are also called seed URLS) which are stored in queue like data structure. It works by downloading pages associated with these URLs extract any HYPERLINKS present in page and iteratively downloading the web pages identified by these HYPERLINKS. Basically after extraction of URLs, it enqueue it. There are various strategies it can either use breadth first, depth first or various other strategies.

2. Search Engine

A search engine is a software system which is used for searching and extracting information (processed data or useful data) from World Wide Web. It works by searching some specified Keywords (which are passed as a query) in documents and return a list of documents satisfying that query. The information which it extract can be in the form or web pages, images, video, pdf files or various other type of files. Some search engine also mine data from databases (is a well organized and systematic collection of data which can be easily accessed, managed) or open directories. Example Google, Ask, Bing, Yahoo Alexa, AltaVista. There are various components of Web search engine.

- Web Crawler
- Databases
- Search Interface
- Search Algorithm
- Ranking Algorithm

A. Web Crawler

It is also known as Spider or Bots. It is a component which traverses the web to download pages.

B. Databases

All the information or pages that the web crawler gets from web is stored in databases. It is also called repository. Every time when we use the search engine, it is the database we are searching

C. Search Interface

It is an interface between user and databases. It helps the user to retrieve information from databases/repository by interacting through the search interface.

D. Search Algorithm

Each search engine interprets the terms you enter in to search box through search interface in different ways by the use of search algorithm. Most search engines allow the use of various operator like and, or, not operation and phrases(which generally use quotation mark around the phrase).

E. Ranking Algorithm

This is component which is used to rank most important pages. Importance of pages can be calculated on various basis like inlink, term frequency, clickthrough analysis(means dropping of some high ranking pages that are not attracting clicks, while promoting lower ranking pages that do pull in visitor). The figure 1 shows the basic architecture of Web Search Engine.

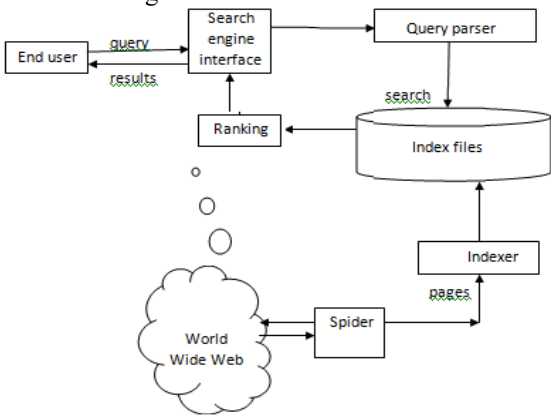

Figure 1: Web Search Engine Architecture

3. Search Engine Working

A Search engine maintains the following processes:
- Web Crawling
- Indexing
- Searching

4. General Architecture of Web Crawler

The basic working of web crawler in Web Search Engine can be described as:
1. Select starting URLs (Seed URLs)
2. Add it to processing queue
3. Now extract URLS from processing queue
4. Download the web page corresponding to that URL
5. Parse the web page and extract the Hyperlinks as new URLs
6. Add all the newly extracted HYPERLINKS (if it is not present already) into the processing queue.

Go to step 2 and repeat while processing queue is not empty.

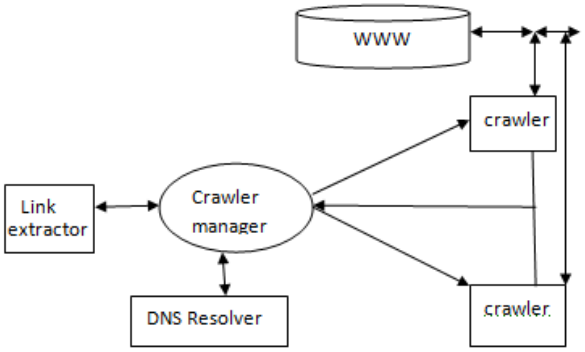

Figure 2: Working process of Web Crawler

5. Classification of Web Crawler

There are various parameters on the basis of which Web Crawler can be classified:
- Coverage area
- Domain Specific
- Mobility
- Load distribution

A. Based on coverage area

Coverage Area means Web Area. On the basis of web area covered by crawler we can classify into two categories:
- Focused Crawler
- Unfocused Crawler

a) *Focused Crawler:* It attempts to focus only on certain types of pages i.e. pages on certain topic and avoiding irrelenant web regions to eliminate irrelevant data.
b) *Unfocused Crawler:* It searches over the entire web to retrieve the pages and construct the index. It deals with a database of large dimension.

B. Based on domain specific

This type of crawler traverses the web on the basis of specific domains.
- Topic specific crawler
- Semantic /ontologies based crawler

a) *Topic Specific Crawler*: It is used for searching information related to specific topic.It selects and retrieves only relevant pages. For each URL relevance is computed and if it is found to be important then only that URL is added to queue.
b) Semantic/Ontologies Based Crawler: These Crawlers make use of semantics which helps to download only relevant pages. Semantics are usually provided by Ontologies. Ontologies provide a common vocabulary of an area, define meaning of terms and relationships between them.

C. Based on load distribution

In order to increase the coverage and decrease the bandwidth usage, crawlers distribute and localize the load on the basis of which it is categorized as:
- Intra Site Parallel Crawler
- Distributed Crawler

a) *Intra Site Parallel Crawler:* In this all crawling processes run on same local network and communicate through high speed interconnection.
b) Distributed Crawler: In this all crawling processes run on geographically distant location connected by internet.

D. Based on mobility

In order to filter out irrelevant data crawler are transported to site of source. Different classes of mobile based crawler are:
- Mobile Crawler

a) *Mobile Crawler:* It uses mobile agents. A mobile agent is automatic independent program that act on behalf of its owner. Mobile crawler is transported to remote site where they filter out unwanted data locally before transferring it to search engine. It compresses the data before transferring to search engine and reduces the amount of data transferred and bandwidth consumption.

6. Issues with Crawler

A Crawler for a large search engine has to address two main issues:
- It has to have good crawling Strategy i.e. a strategy to decide which page to download next.
- It has to have highly optimized system architecture which can increase the download rate (means maximum number of pages per second) while robust against system crashes and manageable and considerate of resources and servers.

The Crawler Design Issues are:
- How should the collection be updated in batch mode or steady mode
- How should the collection be updated in place or shadowing
- How should crawler get time sensitive information
- How should the crawling process be parallelized
- How should the crawler refresh page
- How should the crawler get relevant pages to query

A. How should the collection be updated in batch mode or steady:

A crawler needs to revisit pages in order to maintain the freshness of repository. It can be done by two ways:

- BATCH MODE
- STEADY MODE

a) *Batch mode:* A batch mode runs periodically (a week, month) updating all pages in collection in each crawl.

b) *Steady mode:* A steady crawler runs continuously without any pause.

Both mode provides same average freshness but steady mode is better than batch because it can collect pages at lower peak speed.

B. How should the collection be updated in place or shadowing

When a crawler updates the old copy of repository it may do it in two ways:

- In inplace crawler updates the page in place in repository
- In shadowing a new set of pages is collected and stored in separate space from the current collection. After collection and processing of new pages old one is replaced by new one.

Shadowing may improve the availability of current collection but it may decrease freshness as compare to in place.

C. How should the crawler get time sensitive information

As previous crawls are not archived so search results pertain only to single copy of recent instant Due to which if user request pages containing past data then search engine will be unable to provide that information because it is not possible to search files that represent the snapshot of web over time.

D. How should the crawling process be parallelized:

Parallelization reduces the work load on single machine and generate a continuous stream of new URLs of documents to be downloaded. A efficient parallel computing technique is required to distribute the URLs to distributed network with less bandwidth consumption, network load and overlap.

E. How should the crawler refresh page

Since web pages are changing at very different rates crawler has to decide which page to revisit and which to skip. If the page is changing rarely the crawler has to revisit less often.

F. How should the crawler get relevant pages to query

With increase of web size, application which are used for processing data are also increasing. The goal of search engine is to provide the user relevant and meaningful information during querying, searching, data extraction, mining. So the main task is to determine when a page is to be present in a collection are related to page contents e.g. words, phrases. There can be situations in which inner structure of pages provide better criteria to define a collection than their contents.

7. Parallel Crawler

Parallel crawler is a crawler in which multiple processes are run in parallel in which each process performs the same task as of single crawler to maximize the download rate.

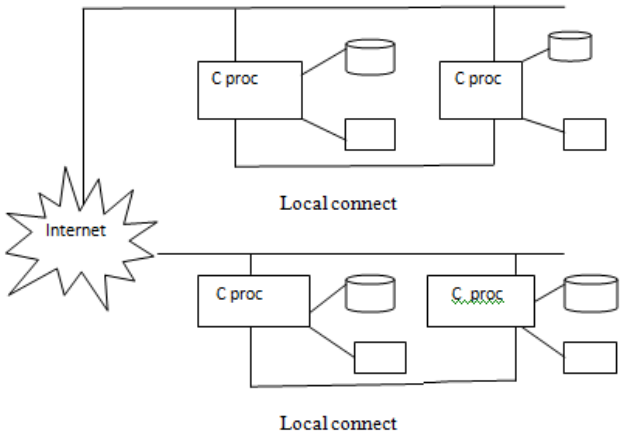

Figure 3: Working of Parallel Crawler

A. Issues

As every technology has some pros and cons, there are also some issues associated with this crawler. These are as follows:

- Overlap
- Communication Bandwidth
- Quality

a) ***Overlap:*** When more than one process run in parallel there can be possibility that same page is downloaded multiple times, This decreases the crawler`s effectiveness. In this one process may not be aware that another process has downloaded the page.

b) ***Communication Bandwidth:*** To reduce the overlap and improve the quality processes need to communicate with each other. This consumes a lot of network/communication bandwidth

c) ***Quality:*** As in parallel crawler each process is unaware of other process download, so it may not be aware about complete image of web that they have collectively downloaded so far. Due to this each process make download decision (which important page to download next) only based on its own image and make poor crawling decision. Then how can we make sure that quality of downloaded pages is as good as for parallel crawler as for centralized one.

B. Advantages of Parallel Crawler

There are various advantages of parallel crawler:

- Scalability
- Network Load Dispersion
- Network Load Reduction

8. Related Work

A. Parallel Crawler [1]

It work gives the complete introduction of parallel crawler, its architecture, issues, application It describes the various ways of coordination among parallel processes (like independent, static, dynamic), reducing overlap (by firewall mode, exchange mode) and increasing coverage. It also describes use of inter partitions link to increase coverage, various method of partitioning the web (URL hash based, Site hash based, hierarchical), URL exchange minimization technique (batch communication, replication) and various

evaluation models (overlap, communication overhead, quality, coverage).

B. Search optimization technique for Domain Specific Parallel Crawler [2]

This paper gives general introduction of various types of crawler. In this paper a technique is described to reduce the load of web and optimize the parallelization by assigning specific domain to each crawler process (by distributing the different domains). It also describes the technique used for optimizing the search operation on WWW which increases efficiency of search engine and providing relevant information frequently by use of Selection factor algorithm (which helps in increasing the rate of downloaded documents in which user has interest)

C. An architecture of parallel crawler based on Augmented Hypertext documents [3]

It describes the WWW as client server architecture. This paper explains the basic working of crawler and gives brief introduction to architecture of google search engine and Mercator. It explains the design of Parallel crawler based on augmented Hypertext documents which emphasis on construction of TOL (table of links) which contains the links in document in the form of meta data to increase the downloading rate. It divides the information retrieval system in to two components:

- Crawling system
- Hypertext documents system

Crawling system is divided in to mapping process and crawling process. Mapping process resolves the IP address and crawling process downloads and process documents.
Hypertext document system provides a TOL along with each document to be downloaded. This paper describes the applications and advantage of Parallel crawler based on augmented hypertext documents. It also describes efficiency of DF (Document Fingerprint) and.TVI (table of variable information). TVI file is used to update the main document by storing fresh information.

D. An Extended model for Effective migrating parallel Web crawling with domain specific and Incremental crawling [5]

This paper describes the architecture of web crawler which combines the advantage of all three crawlers:

- Incremental
- Domain Specific Parallel
- Migrating

Incremental feature maintains the freshness, parallel nature reduce the work load, domain specific increase quality, and migrating behavior migrates the crawler process to host machine by which crawling process runs locally instead of Search engine repository to reduce the communication overhead. This paper describes the architecture of migrating web crawler, its type (intra site and distributed), its advantages, issues associated with it, comparison of various crawler in terms of flexibility robustness and role of central coordinator system, and future scope of migrating domain specific parallel incremental crawler

E. A Novel Architecture for Topic specific web crawler[6]

This paper defines novel architecture of parallel crawler which crawls the pages based on some specific topic. This architecture reduces the work load, increases scalability and provides load sharing. It assigns the domain on the basis of internal and external URLs such that internal URL belong to domain. It overcomes the limitations of firewall mode by distributing external URLs to topic specific crawler identified by topic manager.

9. Comparison Table

This table compares the previous research paper on basis of evaluation model parameters of parallel crawler.

Table 1: Comparison result of reviewed papers

DESIGN ISSUES	[1]	[2]	[3]	[5]	[6]
Coverage	good	less	good	less	Less
Overlap	less	more	moderate	less	more
Quality	good	good	good	good	good
Communication overhead	less	high	less	high	high

10. Conclusion

As the size of web is growing it becomes difficult for a single process to retrieve the large portion of web. That is why it is necessary to parallelize the process to increase the performance like scalability download rate. This paper defines the introduction of parallel crawler, issues, technologies proposed for parallel crawler, their comparison

References

[1] Junghoo cho, Hector Molina," parallel crawler".
[2] Anita Saini, Vinit Kumar, Nidhi Tyagi, "Search optimization technique for Domain Specific Parallel Crawler ".
[3] A.K.Sharma, J.P.Gupta, , D.P.Agarwal, " An architecture of parallel crawler based on Augmented Hypertext documents "
[4] Dr Rajender Nath, Khyati Chopra "Web Crawlers: Taxonomy, issues and challenges".
[5] Md. Faizan Farooqui1, Dr. Md. Rizwan Beg, Dr. Md. Qasim Rafiq " An Extended model for Effective migrating parallel Web crawling with domain specific and Incremental crawling ".
[6] Navita, Mahesh "A Novel Architecture for Topic specific web crawler".