International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 6 Issue: 12 33 - 36

Analysis on Web Crawling Algorithms

Deepak Mayal
Department of Electronics and Communication
AIACTR-GGSIPU
New Delhi, India
mayal.deepak97@gmail.com

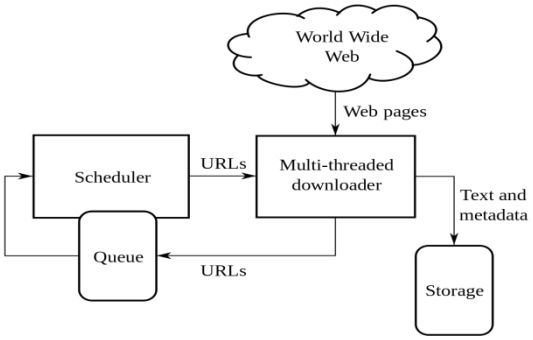
Abstract: World Wide Web (WWW)also referred to as web acts as a vital source of information and searching over the web has become so much easy nowadays all thanks to search engines google, yahoo etc. A search engine is basically a complex multiprogram that allows user to search information available on the web and for that purpose, they use web crawlers. Web crawler systematically browses the world wide web. Effective search helps in avoiding downloading and visiting irrelevant web pages on the web in order to do that web crawlers use different searching algorithm . This paper reviews different web crawling algorithm that determines the fate of the search system.

Keywords: *WebCrawler, Web Search, Web Crawling, Web crawling algorithms*

*****

I. Introduction

Web is a dynamic entity with new pages/data getting added, updated every day so there is a regular need for users as well as applications to stay updated, this where web crawler comes into play. A web crawler is an automated program that scans through different internet pages to search web data. A simple web crawler starts with a base URL often known as seed URL. As it visits the base URL it gathers all the links, hyperlinks available in that page and adds them to a URL list also known as crawl frontier. Next, it picks a URL from the frontier and repeats the same process repetitively until the suitable page is found or higher level objective is reached. Below is the architecture of web crawler



The Web Crawler make use of different searching algorithm in order to crawl the web some of which are discussed in this paper.

II. Web Crawling Algorithms

A. Breadth First Search Algorithm

This algorithm starts with a base URL and searches for all the neighboring URLs that are at the same level. If the desired URL is found it returns success, if not then searching continues and it proceeds to the next URL in the same level until the goal is reached. If no more URLs are present in the same level then it proceeds to the next level. When all the URLs are processed but the appropriate URL is still not found then it returns failure. This algorithm is easy and simple as compared with others algorithm.

The breadth First Search will crawl in A-B-C-D-E-F manner.

Breadth First Algorithm is well suited for cases where the objective is found on the shallower parts in a deeper tree. The major limitation with this algorithm is that it does not perform much well when the branches are many in a game tree, especially chess game.

B. Depth First Algorithm

This algorithm starts with a root URL and traverses in depth through each child URL rather than searching URL at the same level. In this, if more than one child are there then the leftmost child is given high priority and it traverses deep down until no more child can be found. Here backtracking is used to process the next unvisited node.

The depth First Search will crawl in A-B-D-C-E-F manner.

The depth-first algorithm is well suited for search problems, but if the branches are large it might end up in the infinite loop.

C. Page Rank Algorithm

This algorithm determines the relative importance of the web pages in any website by calculating page rank of each given page. The page rank is calculated by relatedness between web pages. The page rank depends on the rank of all pages that are linked to it which are referred as inbound link. If the page is linked by many pages having high rank than it is considered of more importance compared to other websites. Therefore, Page rank of web page can be described as summation of weight of all input links. Initially, all pages are given equal page rank i.e 1/n(n=number of pages) and then the page rank of each page is calculated as:

PR (P) = (1-d) + d (PR(X1)/C(X1)) +…+
PR(Xn)/C(Xn)

where, PR (P) = Page Rank of a page P,

d = Damping factor in range 0 to 1,

PR(Xi)=Page Rank of page Xi which links to page A,

C(Xi)=Number of outbound links on a given Xi page.

The disadvantage with Page Rank Algorithm is that older pages may have high rank compared to new pages even if the new page has updated and excellent content . Also this algorithm is unable to handle natural language queries as results are solely based on keywords provided not on the meaning of the query.

D. Path-ascending crawling algorithm

This algorithm crawls each path from the home to the last file of that URL thus helping the crawler to extract more information from that URL. For example, when given a URL of http://rohit.org/hamster/index, it will attempt to crawl /hamster/index, /hamster/, and /.

The main advantage of this algorithm is that it is effective in finding resources for which no inbound link would have been found in regular crawling.

E. Naive Best-First Crawling Algorithm

This algorithm represents a crawled web page as a vector of words weighted by occurrence frequency. In this, the crawler puts URL in the crawler frontier i.e priority queue based on the cosine similarity of the page with that of the query provided by the user. Then at each repetition crawler picks a URL from the queue and returns unvisited URLs depending on the similarity scores with the parent page. The cosine similarity result is based on four attributes URL words, anchor text, parent text and the surrounding text of an URL.

The cosine similarity between between the page aand a query bis calculated as

$$sim(b,a) = \frac{v_b . v_a}{\|v_a\| \|v_b\|}$$

where v_a and $v_b$are the frequencyvectors, $v_a.v_b$ is the dot product , and ||v|| is the Euclidean form of the vector v.

F. SHARKSEARCH

SharkSearch uses the same similarity measure to calculate the relevance of the URL as used by the naive-best first algorithm. However, sharksearch have a much better potential score for the links. The anchor text, link-context, and inherited score from parents/ancestors influence the potential scores of links. It maintains depth bound that is if irrelevant pages are being crawled in a given direction then crawler stops crawling in that direction. Each URL is associated with two values depth bound which is given by the user and the potential score which is calculated as:

Score(URL)= y*inherited(URL)+
(1-y)*neighbourhood(URL)
where y<1 is a parameter

$$\text{Inherited(URL)} = \begin{cases} \mu * sim(q,p) \text{ if } sim(q,p) > 0 \\ \mu * inherited(p) \end{cases}$$

where μ <1 is a parameter

neighbourhood(URL)= β *anchor(URL)+(1- β)*
context(URL)

Where β <1 and anchor score is the similarity between query to that of anchor text. Context score broadens the context to include nearby words for better results. Thus resulting augmented context Is used for calculating the score

$$\text{Context(url)} = \begin{cases} 1 \text{ if anchor(url)} > 0 \\ \text{sim(q, aug\_context)} \end{cases}$$

G. Semantic Web crawler Algorithm

In semantic web crawling algorithm when a query is made by the user, the query is refined through a special query processor which basically removes stop words and also does stemming of the query words to get more precise search results. Special query processor consists of the following significant components.

1) Stopword identification Module: Stop words are basically prepositions which are needed to be filtered out because they make poor searching terms. Removing Stop words is the firstl step towards query processing technique.

Sample S={India is beauitiful}

On applying Stop words Removal process

Sample S'={India beautiful}

2) Stemming Module: Next step includes refining of the query so that search result is effective and much more expressive. For eg, a word "do" in the query is refined to "does", "done".

Both of these module help in getting more accurate keywords, thus with help of lexical database much more accurate sense is obtained.

3) Lexical Database: Lexical Database is basically a large collection of Synonyms, Antonym, Meronym, Holonyms of English words.

4) Crawler: Crawler fetches metadata and determine its sense using lexical database. After that the query is refined and its sense is also determined using the same lexical database. Atlast the senses of both are compared and the result varifies if they match or not.

H. Online Page Importance Calculation Algorithm

This algorithm is similar to that of page rank algorithm. In this rather assigning page rank to each page all pages are given cash value. Pages having higher cash value are first downloaded and at each stage cash is distributed among the pages it points when a page is downloaded. Initially, all pages are assigned same cash value i.e 1/n.

The cash value is calculated in one step and in a very short duration of time. The main limitation of this algorithm is that a single page would be downloaded many time and that will increase response time for our crawler.

I. HITS Algorithm

Hyperlink-Induced Topic Search is a link analysis algorithm which uses score to calculate the relevance of the webpage. For that it calculates two values that are authority value (estimates the value of the content of the page) and hub value(estimates the value of its links to other pages).

Steps involved in HITS Algorithm:

1. On basis of search query HITS assembles relevant pages via text-based search algorithm. This set is called the root set and is combined with all the web pages that are linked from it and pages that are linked to it

2.HITS assigns each page with hub weight h(p) and authority weight that is initialized as 1.

3. HITS then repetitively update hub and authority weight.

Authority is updated by:

$$auth(a) = \sum_{j=1}^{n} hub(j)$$

where n is the total number of pages connected to a and j is a page connected to p.

Hub score is given by:

$$hub(a) = \sum_{j=1}^{n} auth(j)$$

where n is the total number of pages to whichaconnects and j is a page which a connects to.

III. CONCLUSION

The main aim of this paper was to discuss different web crawling algorithms along with their advantages and disadvantages. I believe that all of the algorithms discussed in this paper are effective for web search but the advantages favor more for sharksearch algorithm due to the use of inherited score, thus preferring the children of a node that has better score and also making use of meta-information contained in the links to calculate the potential score.

REFRENCES

[1] Rashmi Janbandhu, Prashant Dahiwale, M.M.Raghuwanshi “Analysisof Web Crawling algorithms” International Journal on Recent andInnovation Trends in Computing and Communication, Vol. 2

[2] Apoorv Vikram Singh , Vikas , Achyut Mishra “A Review of Web Crawler Algorithms”International Journal of Computer Science and Information Technologies, Vol. 5 (5) , 2014, 6689-6691

[3] Pavalam S M, S V Kashmir Raja, Felix K Akorli and Jawahar M “ASurvey of Web Crawler Algorithms” International Journal of ComputerScience Issues, Vol. 8

[4] http://pi.math.cornell.edu/~mec/Winter2009/RalucaRemus/Lecture4/lecture4.html

[5] Svapnil V. Patil, Sharmila M. Shinde “*Ontology Based semantic web Crawler Mechanism for Information Discovery* ” International Journal of Advance Research

inComputer Science and Management Studies Volume 2, Issue 12

[6] Yongbin Qin and Daoyun Xu "A Balanced RankAlgorithm Based on PageRank and Page Beliefrecommendation"

[7] Rahul kumar, Anurag Jain and Chetan Agrawal "SURVEY OFWEB CRAWLING ALGORITHMS"Advances in Vision Computing: An International Journal (AVC) Vol.3, No.3

[8] Carlos Castillo , Mauricio Marin , Andrea Rodriguez, —Scheduling Algorithms for Web Crawlinginthe proceedings of WebMedia and LA-Web, 2004.

[9] Andas Amrin*, Chunlei Xia, Shuguang Dai "**Focused Web Crawling Algorithms**"

[10] Abhinav Garg, Kratika Gupta* and Abhijeet Singh " Survey of Web Crawler Algorithms" International Journal of Advanced Research in Computer ScienceVolume 8, No. 5

[11] Carlos Castillo, Mauricio Marin, Andrea Rodriguez, and Ricardo Baeza-Yates."Scheduling algorithms for Web crawling".In Latin American Web Conference (WebMedia/LA-WEB), RiberaoPreto, Brazil, 2004. IEEE Cs. Press

[12] Mehdi Ravakhah, M. K. "Semantic Similarity BasedFocused Crawling" 'First International Conference on Computational Intelligence, Communication Systems and Networks', 2009

[13] https://www.wikipedia.org/