INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS
ISSN 2320-7345

A SURVEY ON WEB FOCUSED INFORMATION EXTRACTION ALGORITHMS

Satwinder Kaur[1] & Alisha Gupta[2]

[1]Research Scholar (M.tech CSE), [2]Assistant professor
HEC, Jagadhri, Kurukshetra University, Haryana, India.

Abstract: - The World Wide Web is the largest collection of data today and it continues increasing day by day. A web crawler is a program from the huge downloading of web pages from World Wide Web and this process is called Web crawling. To collect the web pages from www a search engine uses web crawler and the web crawler collects this by web crawling. Due to limitations of network bandwidth, time-consuming and hardware's a Web crawler cannot download all the pages, it is important to select the most important ones as early as possible during the crawling process and avoid downloading and visiting many irrelevant pages. This paper reviews help the researches on web crawling methods used for searching.

Keywords: - Web crawler, Web Crawling Algorithms, Search Engine

1. Introduction

A web crawler or spider is a computer program that browses the WWW in sequencing and automated manner. A crawler which is sometimes referred to spider, bot or agent is software whose purpose it is performed web crawling. The basic architecture of web crawler is given below (Figure1). More than 13% of the traffic to a web site is generated by web search [1]. Today the size of the web is thousands of millions of web pages that is too high and the growth rate of web pages are also too high i.e. increasing exponentially due to this the main problem for search engine is deal this amount of the size of the web. Due to this large size of web induces low coverage and search engine indexing not cover one third of the publicly available web [6].By analyzing various log files of different web site they found that maximum web request is generated by web crawler and it is on an average 50% [1]. Crawling the web is not a programming task, but an algorithm design and system design challenge because of the web content is very large [2]. At present, only Google claims to have indexed over 3 billion web pages.About 40% web pages change weekly when we consider lightly change, but when we consider changing by one third or more than the changing rate is about 7% weekly

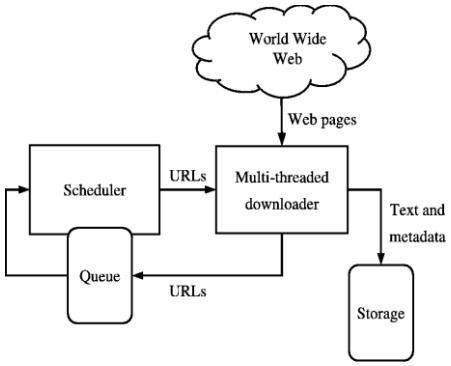

Fig 1 Architecture of web crawler

2. BASIC CRAWLING TERMINOLOGY:

Before discussing the architecture of a crawler, it is worth to explain some of the terminology that is related with crawlers.

- **SEED PAGE:** By crawling, we mean to traverse the web by recursively following links from a starting URL or a set of starting URLs. This starting URL set is the entry point through which any crawler starts by searching procedure. This set of starting URL is known as "Seed Page". The selection of a good seed is the most important factor in ay crawling process.
- **FRONTIER (Processing Queue):** The crawling method starts with a given URL(seed), extracting links from it and adding them to an un-visited list of URLs. This list of un-visited links or URLs is known as, "Frontier". Each time, a URL is picked from the frontier by the Crawler Scheduler. The maintenance of the Frontier is also a major functionality of any crawler.
- **PARSER:** Once a page has been fetched, we need to parse its content to extract information that will feed and possibly guide the future path of the crawler. Parsing may imply simple hyperlink/ URL extraction or it may involve the more complex process of tidying up the HTML content in order to analyze the HTML tag tree. The job of any parser is to parse the fetched web page to extract list of new URLs from it and return the new un-visited URLs to the frontier.

3. WEB CRAWLER STRATEGIES

3.1 Breadth First Search Algorithm: According to this strategy, search is started from seed page and continues to all the immediate neighboring pages. Only after crawing all the first level web pages, it moves to the next level web pages.[7]

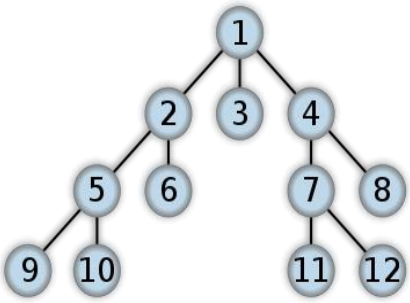

Fig 2 Breadth First Search Algorithm

3.2 Depth First Crawling Algorithm: In this strategy, search is done across the depth of web graph. Only after reaching the deepest link after which no link is present, the neighboring pages can be crawled. [1]

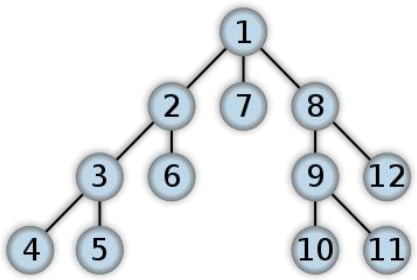

Fig 3 Depth First Search Algorithm

3.3 Page Rank algorithm: This algorithm determines the importance of web pages based on a pagerank metric. It state that if a page has important links to it , its link to other pages also contribute to their importance and a page with high pagerank is most relevant page to be downloaded. The pagerank of a page A is given by:

PR(A)= (1-d)/|D| + d(PR(T1)/C(T1)+….+PR(Tn)/C(Tn))

Where

PR(A) = pagerank of page A

T1....Tn= inlinks to page A

C(A) = no. of links going out of page A

D = set of all web pages

d= damping factor which is often assumed to 0.85

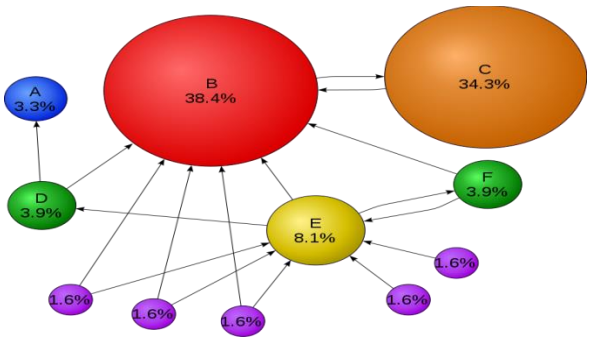

Fig 4 Page Rank Algorithm

3.4 Weighted pagerank algorithm: This algorithm is an improved version of pagerank which assigns more value to more important pages instead of dividing the rank value of a page evenly among all its outgoing links [13]. The weighted pagerank is thus given by:

PR(u)= (1-d)/|D| + d(PR(V1)Win(V1,u)Wout(V1,u)+….+PR(Vn)Win(Vn,u) Wout(Vn,u)

Where Win(v,u) = weight of link(v,u) calculated based on the number of inlinks of page u and the number of inkinks of all reference pages of page v

Wout(v,u) = weight of link(v,u) calculated based on the number of outlinks of page u and the number of inlinks of all reference pages of page v

PR(u)=Pagerank of page u

3.5 Rankmass crawling algorithm: This paper deals with the two important issues of crawler (1) coverage to cover most of the web and (2) efficiency by giving high priority to important pages. This work was important as it is costly to house a large corpus of web pages. As the web is very wide the efficiency of search result measured by search engine indexing is sometime misleading [12]. So this algorithm was devised to provide guarantee that a crawler has downloaded the most important part of web before it stop crawling. The rankmass metric, a variant of personalized pagerank was proposed for comparing the quality of search engine indexes. Their crawler was focused on the user relevant pages and can prioritize the crawl downloading high personalized pagerank first and ultimately high rankmass is achieved when the crawl is over. To determine the importance of pages their pagerank was computed using personalized pagerank that assumed that a user goes to a trusted site rather than to a page of equal probability. So the pagerank of page A is then defined as

PR(Ai)= (1-d)(Ti) + d(PR(R1)/C(R1)+….+PR(Rn)/C(Tn))
Where
PR(A) = pagerank of page A
R1....Rn= inlinks to page A
Ti = trust score of page i
C(A) = no. of links going out of page A
d= damping factor which is often assumed to 0.85

4. RESEARCH SCOPE

As, the defined concepts for web crawling and improving its performance by the various crawling algorithms have been explained here. It has not end of the work for improving performance of crawling. There are many more techniques and algorithms may be considered for crawler to improve its performance. We can also improve its performance by comparing two metrics i.e. rankmass coverage metric and weighted pagerank metric. By weighted pagerank metric, we will design an improved rankmass crawling algorithm

5. CONCLUSION

The paper surveys several crawling methods or algorithms that are used for downloading the web pages from the World Wide Web. We believe that all of the algorithms discuss in this paper are well effective and high performance for web search, reduce the network traffic and crawling costs, but overall advantages and disadvantage favor more for By using HTTP Get Request and also Dynamic Web Page and download updated web pages By the using of filter is produce relevant results.

5. REFERENCES

[1] Pavalam S M, S V Kashmir Raja, felix K Akorli and Jawahar M, A Survey Of web Crawler Algorithms, International Journal of Computer Science issues, Vol. 8, Issue 6, No. 1, November 2011.
[2] S. Brin and L. Page. The anatomy of a large-scale hyper textual Web search engine. Computer Networks and ISDN Systems, 30(1{7):107{117, April 1998.
[3] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. Journal of the ACM, 46(5):604–632, September 1999.
[4] J. Cho, H. Garcia-Molina, and L. Page. Efficient crawling through URL ordering. Computer Networks and ISDN Systems, 30(1-7):161–172, 1998.
[5] R. Baeza-Yates, C. Castillo, M. Marin, and A. Rodriguez. Crawling a country: better strategies than breadth-first for web page ordering. In WWW ’05: Special interest tracks and posters of the 14th international conference on World Wide Web, pages 864–872, 2005.
[6] S. Abiteboul, M. Preda, and G. Cobena. Adaptive on-line page importance computation. In WWW ’03:

Proceedings of the 12th international conference on World Wide Web, pages 280–290, New York, NY, USA, 2003

[7] M. Najork and J. L. Wiener, Breadth first crawling yields high-quality pages, In Proceedings of the Tenth Conference on World Wide Web, pages 114{118, Hong Kong, May 2001.

[8] Dennis Fetterly, Nick Craswell, Vishwa Vinay, The impact of Crawl Policy On web Search Effectiveness in Proceeding of SIGIR, july 2009.

[9] S. Pandey and C. Olston. User-centric web crawling. In Proc. 14th WWW, pages 401–411, 2005.

[10] S. Pandey and C. Olston. Crawl ordering by search impact. In Proceedings of WSDM, pages 3–14, 2008

[11] M. Henzinger, A. Heydon, M. Mitzenmacher, and M. Najork. Measuring index quality using random walks on the Web. COMPUT. NETWORKS, 31(11):1291–1303, 1999.

[12] J. Cho and U. Schonfeld. Rankmass crawler: a crawler with high personalized PageRank coverage guarantee. In Proceedings of VLDB, pages 375–386, 2007.

[13] Wenpu Xing and Ghorbani Ali, Weighted pagerank algorithm, In Proceeding of the second annual conference on Communication Networks and Services Research(CSNR' 04),IEEE,2004

[14] Neelam Tyagi and Simple Sharma, Weighted Page Rank Algorithm based on number of visits oflinks of web pages, International Journal Of Soft Computing and Engineering (IJSCE) ISSN: 2331-2307, Vol.2, Issue-3, July 2012.

[15] R. Baeza-Yates and C. Castillo. Crawling the infinite web: five levels are enough. Journal of Web Engineering, 6(1):49–72, 2007.