ISSN:0975-9646

A Review of Web Crawler Algorithms

Apoorv Vikram Singh , Vikas , Achyut Mishra

Dept. of Computer Science & Engineering

MNNIT , Allahabad , India

Abstract: **The web today contains a lot of information and it keeps on increasing everyday. Thus, due to the availability of abundant data on web, searching for some particular data in this collection has become very difficult. Emphasis is given to the relevance and robustness of data by the on-going researches. Although only relevant pages are to be considered for any search query but still huge data needs to be explored. Another important thing to keep in mind is that usually one's need may not be desirable for others. Crawling algorithms are thus crucial in selecting the pages that satisfy the user's need. This paper reviews the researches on web crawling algorithms used for searching.**

***Keywords*:-** Web Crawling Algorithms, Crawling Algorithm Survey, Search Algorithms

I. INTRODUCTION

Web search is currently generating more than 13% of the traffic to the websites[12]. The main problem which the search engines have to deal with is the huge and continuously growing Web, which currently is in order of thousands of millions of pages. Because of this large size, no search engine indexes more than one third of the publicly available Web.

When a data is searched, hundreds and thousands of results appear. The users are not persistent enough to go through each and every page listed. So the search engines have a bigger job of sorting out the results, in the order of interest to the user within the first page of appearance and a quick summary of the information provided on a page[1].

Web crawlers are programs which traverse through the web, searching for the relevant information using algorithms that narrow down the search by finding out the closest and relevant information. Researchers are developing scheduling policy for downloading pages from the Web which guarantees that even if all the pages are not downloaded, the most important one's will be downloaded.

II. FUNDAMENTALS OF WEB CRAWLER

A web crawler (also known as a web spider or web robot) is a program or automated script which browses the World Wide Web in a methodical, automated manner searching for the relevant information using algorithms that narrow down the search by finding out the closest and relevant information. The relevance of the information is determined by the algorithm used by the crawler by considering factors like frequency and location of keywords in the web pages. Crawlers also perform the function of fetching new and recently changed websites, and indexing them.

The crawling process generally starts with a set of URLs(Uniform Resource Locator) called the Seed URLs. The crawler visits each of these websites and detects links present on these websites. These links are of two types, the first one being the links to some new web pages which were not crawled before and the second type being the links of those web pages which were modified after the crawl and thus the crawler needs to visit them again. These links are added to the list of links to be crawled. It also notes whether there is any new website or website that has been recently changed (updated), websites that are no more in use and accordingly updates the index. The indexer compiles the list of words it sees and its location on each page for future consultation. The information compiled is mostly because crawlers are mostly text based[1].

A. How the targets are selected ?

The size of the web is huge and is increasing every second, thus it is practically not possible to cover all the websites for a particular search entry. And another problem is that the complete web is not indexed, only a certain percentage of it is indexed. A web crawler always downloads web pages in fraction. Thus for getting relevant pages in first few downloads there is a need for prioritising Web pages. The relevance of a page depends on certain factors like the number of visits on the page. Different strategies such as depth first, breadth first, page rank method are used by different researchers for selecting the websites to be downloaded.

B. Where to start ?

We can start from any seed URL, but this thing should be kept in mind that the starting URL will not reach all the web pages. Another factor to be taken care of is that the pages referenced by the seed URLs should not reference it back to them, else it will restart the crawl. It is always better to have a good seed URL – pages that have been submitted to them by majority of users around the world. For example yahoo or Google can be used to get seed URL by simply entering the keywords into them and considering their resulting links as our seed URLs. This is because these are amongst the popular search engines whose results are prominent and accepted by majority of users around the world[2].

C. Any restrictions on the number of Links to follow

The Web keeps on getting bigger and there is a cost associated with crawling, indexing and storing the results, thus only relevant pages are required to be downloaded. Thus, for this purpose scheduling strategies are needed to minimise crawling time and to reduce the cost and these strategies differ from one search engine to another. As the web is huge and to download as many pages as possible, paral-

lel crawlers are distributed so that multiple downloads can be carried out in parallel[3].

D. Freshness of a page and revisiting policy

The freshness, newness and revisiting of a page also has significant importance while crawling the web so that user is benefited by updated and latest information. Two types of visiting policies have been proposed –Uniform change frequency - the revisiting is done uniformly regardless of its change and Non-uniform change frequency – the revisiting is not uniform and the revisiting is done more frequently and the visiting frequency is directly proportional to the change frequency[4].

IV. WEB CRAWLER STRATEGIES

A. Breadth first search algorithm

This algorithm aims uniform search across the neighbouring URLs present at the same level. This algorithm starts at the root URL and searches all the neighbouring URLs at the same level. If the goal is reached, then it is reports success and the search terminates. If it is not, search proceeds down to the next level, sweeping the search across the neighbouring URLs at that level and so on until the goal is reached. When all the URLs are searched, but the objective is not met then it is reported as failure.

Breadth first search algorithm will be more suitable for applications and situations where the desired results can be obtained in the upper(shallow) levels of a deeper tree. Its performance will get affected if the results will be found in deeper levels. It will not perform so well in problems like game tree where there are so many branches and all the path lead to same objective with the same length of path.

Andy yoo et al [9] proposed a distributed BFS for numerous branches using Poisson random graphs and achieved high scalability through a set of clever memory and communication optimisations.

B. Depth first search algorithm

In this search algorithm we start at the root URL and traverse deeper through the child URL. If there is more than one child, then priority is given to the left most child and traverse deep until no more child is available. It is backtracked to the next unvisited node and then continues in a similar manner [5].

This algorithm makes sure that all the edges are visited once in every breadth [6]. It is well suited for search problems, but when the branches are large then this algorithm might end up in an infinite loop [7].

C. Page rank algorithm

In this algorithm, a certain value called Page Rank is assigned to the pages and this Page Rank is the measure of relevance of that page determined by counting the number of citations and backlinks to that page. The Page Rank of a given page is calculated as

$$PR(A) = (1-d) + d(PR(T1)/C(T1) + \ldots + PR(Tn)/C(Tn))$$

PR(A) : Page Rank of a given Page

d : Dumping Factor

Ti : links

In order to find the Page Rank for a page, called PR(A), we need to find all the pages that link to page A and Out Link from A. We find a page T1, which has link from A then page C(T1) will give no. of Outbound links to page A. We do the same for T2, T3 and all other pages linking to Main page A – and Sum of the values will provide Rank of the web page.

Tian Chong [8] proposed a new type of algorithm of page ranking by combining classified tree with static algorithm of PageRank, which enables the classified tree to be constructed according to a large number of users' similar searching results, and can obviously reduce the problem of Theme-Drift, caused by using PageRank only, and problem of outdated web pages and increase the efficiency and effectiveness of search.

J.Kleinberg [10] proposed a dynamic page ranking algorithm. Shaojie Qiao [11] proposed a new page rank algorithm based on similarity measure from the vector space model, called SimRank, to score web pages. They proposed a new similarity measure to compute the similarity of pages and apply it to partition a web database into several web social networks (WSNs)

D. Path-Ascending Crawling AlgorithmA

This algorithm crawls each path from the home to the last file of that URL. This nature of the crawler helps to get more information from that site. In the above way a crawler ascends to every path in each URL (Uniform Resource Locator) that it intends to crawl. For example when given a seed URL of http://apoorv.org/vikas/achyut.html , it will attempt to crawl /apoorv.org/, /vikas/ and /achyut.html.

The advantage with Path-ascending crawler is that they are very effective in finding isolated resources, or resources for which no inbound link would have been found in regular crawling[13].

E. Focused crawling algorithm

The significance of a page for a crawler can also be expressed as a function of the similarity of a page to a given query. In this approach we can intend web crawler to download pages that are similar to each other, thus it would be called focused crawler or topical crawler[14].

The main thing to be kept in mind is that the page is downloaded only after envisaging the similarity of the text to the given page. The features such as URL, anchor text which are available without downloading that particular page are used to predict the similarity of unvisited page. Focused crawling usually relies on a general Web search engine for providing starting points i.e. its seed URLs. This type of crawler can be used to have specific type of search engines based on their file types[15].

F. Genetic algorithm

Genetic algorithm is based on biological evolution whereby the fittest offspring is obtained by crossing over of the selection of some best individuals in the population by means of fitness function. In a search algorithm solutions to the problem exist but the technique is to find the best solution within specified time [16]. [21] shows the genetic algorithm is best suited when the user has literally very less

or no time at all to spend in searching a huge database and is also very efficient in multimedia results. While almost all conventional methods search from a single point, Genetic Algorithms always operate on a whole population.

G. Naïve Bayes classification Algorithm

Naïve Bayes algorithm is based on Probabilistic learning and classification. It assumes that one feature is independent of another [17]. This algorithm proved to be efficient over many other approaches [18] although its simple assumption is not much applicable in realistic cases [17]. Wenxian Wang et al [19] proposed an efficient crawler based on Naïve Bayes to gather many relevant pages for hierarchical website layouts. Peter Flach and Nicolas Lachiche [20] presented Naïve Bayes classification of structured data on artificially generated data.

H. HITS algorithm

This algorithm put forward by Kleinberg is previous to Page rank algorithms which uses scores to calculate the relevance [22]. This method retrieves a set of results for a search and calculate the authority and hub score within that set of results. Because of these reasons this method is not often used. Joel C. Miller et al [23] proposed a modification on adjacency matrix input to HITS algorithm which gave intuitive results.

V. CONCLUSION

The main objective of the review paper was to throw some light on the web crawling algorithms. We also discussed the various search algorithms and the researches related to respective algorithms and their strengths and weaknesses associated. We believe that all of the algorithms surveyed in this paper are effective for web search, but the advantages favour Genetic Algorithm due to its iterative selection from the population to produce relevant results and Focused Crawling Algorithm due to its smallest response time.

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
2. Rashmi Janbandhu, Prashant Dahiwale, M.M.Raghuwanshi "Analysis of Web Crawling algorithms" International Journal on Recent and Innovation Trends in Computing and Communication, Vol. 2
3. Marc Nojark, "Web Crawler Architecture" retrieved from http://research.microsoft.com/pubs/102936/EDS-WebCrawlerArchitecture.pdf
4. Carlos Castillo , Mauricio Marin , Andrea Rodriguez, ―Scheduling Algorithms for Web Crawling‖in the proceedings of WebMedia and LA-Web, 2004.
5. Alexander Shen "Algorithms and Programming: Problems and solutions" Second edition Springer 2012, pg 135
6. Narasingh Deo "Graph theory with applications to engineering and computer science" PHI, 2004 Pg 301
7. Ben Coppin "Artificial Intelligence illuminated" Jones and Barlett Publishers, 2004, Pg 77.
8. TIAN Chong "A Kind of Algorithm For Page Ranking Based on Classified Tree In Search Engine" Proc International Conference on Computer Application and System Modeling (ICCASM 2010)
9. Andy Yoo,Edmond Chow, Keith Henderson, William McLendon, Bruce Hendrickson, ÄUmit CatalyÄurek "A Scalable Distributed Parallel Breadth-First Search Algorithm on BlueGene/L" ACM 2005.
10. J.Kleinberg "Authoritative sources in a hyperlinked environment", Proc 9th ACM-SIAM Symposium on Discrete Algorithms, 1998.
11. Shaojie Qiao, Tianrui Li, Hong Li and Yan Zhu, Jing Peng, Jiangtao Qiu "SimRank: A Page Rank Approach based on similarity measure" 2010 IEEE
12. StatMarket. Search engine referrals nearly double worldwide.
13. Pavalam S M, Jawahar M, Felix K Akorli, S V Kashmir Raja ― Web Crawler in Mobile Systems‖in the proceedings of International Conference on Machine Learning (ICMLC 2011),
14. Kim, S. J. and Lee, S. H. "An improved computation of the PageRank algorithm‖ in Proc. of the European Conference on Information Retrieval
15. Debashis Hati, Biswajit Sahoo, A. K. "Adaptive Focused Crawling Based on Link Analysis," 2nd International Conference on Education Technology and Computer (ICETC), 2010.
16. S. N. Sivanandam, S. N. Deepa "Introduction to Genetic Algorithms" Springer, 2008, pg 20
17. Harry Zhang "The Optimality of Naive Bayes" American Association for Artificial Intelligence 2004.
18. Rich Caruana, Alexandru Niculescu-Mizil "An Empirical Comparison of Supervised Learning Algorithms" Proc 23 rd International Conference on Machine Learning, Pittsburgh, PA, 2006.
19. Wenxian Wang, Xingshu Chen, Yongbin Zou, Haizhou Wang, Zongkun Dai "A Focused Crawler Based on Naive Bayes Classifier" Third International Symposium on Intelligent Information Technology and Security Informatics, 2010
20. Peter A. Flach and Nicolas Lachiche "Naïve Bayesian Classification of Structured Data" Machine Learning, Kluwer Academic Publishers
21. S.N. Palod, Dr S.K.Shrivastav,Dr P.K.Purohit "Review of Genetic Algorithm based face recognition" International Journal of Engineering Science and Technology (IJEST) Vol. 3 No. 2 Feb 2011
22. Kleinberg, John "Hubs, Authorities, and Communities" ACM computing survey,1998.
23. Joel C. Miller, Gregory Rae, Fred Schaefer "Modifications of Kleinberg's HITS Algorithm Using Matrix Exponentiation and Web Log Records" Proc. SIGIR'01, ACM 2001