A Survey of Web Crawler Algorithms

Pavalam S M[1], S V Kashmir Raja[2], Felix K Akorli[3] and Jawahar M[4]

[1] National University of Rwanda
Huye, RWANDA

[2] SRM University
Chennai, INDIA

[3] National University of Rwanda
Huye, RWANDA ***Email address***

[4] National University of Rwanda
Huye, RWANDA

Abstract

Due to availability of abundant data on web, searching has a significant impact. On-going researches place emphasis on the relevancy and robustness of the data found, as the discovered patterns proximity is far from the explored. Inspite of their relevance pages for any search topic, the results are huge to be explored. Also the users' perspective differs from time to time from topic to topic. Usually ones' want is others unnecessary. Crawling algorithms are thus crucial in selecting the pages that satisfies the users' needs. This paper reviews the researches on web crawling algorithms used on searching.

Keywords: ***web crawling algorithms, crawling algorithm survey, search algorithms***

1. Introduction

These are days of competitive world, where each and every second is considered valuable backed up by information. Timely Information retrieval is a solution for survival. Due to the abundance of data on the web and different user perspective, information retrieval becomes a challenge .

When a data is searched, hundreds and thousands of results appear. The user's don't have persistence and stretch 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 interestingness of the user within the first page of appearance and a quick summary of the information provided on a page.

Web crawlers are programs which traverse through the web searching for the relevant information [1] using algorithms that narrow down the search by finding out the most closer and relevant information. This process is iterative, as long the results are in closed proximity of user's interest. The algorithm determines the relevancy based on the factors such as frequency and location of keywords.

Web pages needs not only relevance but also authoritativeness – from a trusted source of strong, precise information [2]. Search engines uses algorithms which sorts, ranks the result in the order of authority, that is closer to the user's query. Many algorithms are is in use - Breadth first search, Best first search, Page Rank algorithm, Genetic algorithm, Naïve Bayes clssification algorithm to mention a few.

There are chances that the website may not contain the keyword, but they are completely relevant website. For example if the user is searching for cars, then the result returned are information about used cars for sale rather than information about cars manufacturers website.

Not all information represented are useful. The search engine techniques may become useless or junky if the information it draws are not attracting users, especially if the malicious user who are trying to attract more traffic in to their site by embedding the most used keywords invisibly in to their site. The challenges are relevancy, robustness and the ability to download large number of pages.

2. Fundamentals of web crawling

Crawlers have bots that fetches new and recently changed websites, and indexes them. By this process billions of websites are crawled and indexed using algorithms (which are usually well guarded secrets) depending on a number of factors. Several commerical search engines changes the factors often to improve the search engines process.

It generally starts with a set of URLs from the previous crawl, visits each of these websites, detects links and adds it to the list of links to crawl. 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 index is updated.

The indexer compiles the list of words it sees and its location on each page for future consultation. The information compiled are mostly because crawlers are majoritively text based.

When an user initiates a search, the key words are extracted and searches the index for the websites which are most relevant. Relevancy is determined by a number of factors and also it differes for the different search engines.

2.1 How the targets are selected?

The size of the web is huge, search engines practically can't be able to cover all the websites. Only 60 percentage are the indexed web [3]. There is a high chances of the relevant pages in the first few downloads, as the web crawler always download web pages (in fractions). This calls for measures for prioritizing Web pages. The importance of a page is a function of its essential quality, its reputation in terms of links or visits, and even of its URL. Different researchers used different strategies such as bread firth, depth first, page rank for selecting the websites to be downloaded.

2.2 Where to start?

We have to start from any URL (Seed), but imagine that the starting URL couldn't reach all the web pages or even the pages referenced by seed URL doesn't reference it back, which eventually makes us to restart the crawl. It is always better to have a good seed URL – pages that has been submitted to them by majority users around the world. For example yahoo or Google.

2.3 Any restrictions on the number of pages to follow (Link)

There is a cost associated with crawling, indexing and storing the results. When the web gets bigger and bigger, the "better" pages are downloaded. So there needs to be a scheduling strategy tto minimize crawling time and to reuduce cost [4] and it differs from one search engine to another. As the web is huge and to download as many pages as possible, parallel crawlers are distributed so that mulitple downloads can be carried out in parallel [5]

2.4 Freshness of a page and revisiting policy

When the same copy exists in the local as well as the remote sources, then it is considered to be the "fresh" page. Cho and Garcia [6] calculated the freshness of a page as

$$F(e_i;t) = \begin{cases} 1 \text{ if } e_i \text{ is up-to-date at time } t \\ 0 \text{ otherwise.} \end{cases} \qquad (1)$$

Where e_i is the element of database

And the age of a page as

$$A(e_i;t) = \begin{cases} 0 & \text{if } e_i \text{ is up-to-date at time } t \\ t - t_m(e_i) & \text{otherwise.} \end{cases} \qquad (2)$$

Where tm(ei) is the time of first modification of ei after the most recent synchronization. The freshness drops to zero when the real-world element changes and the age increase linearly from that point on. When the local element is synchronized to the real-world element, its freshness recovers to one, and its age drops to zero.

Two types of visiting policy has been proposed – Uniform change frequency - the revisiting is done at the uniform regardless of its change and non uniform change frequency – the revisiting is not uniform and the revisitng is done more frequently and the visiting frequency is directly proportional to the change frequency.

3. Web crawler strategies:

3.1 Breadth First Search Algorithm:

This algorithm aims in the uniform search across the neighbour nodes. It starts at the root node and searches the all the neighbour nodes at the same level. If the objective is reached, then it is reported as success and the search is terminated. If it is not, it proceeds down to the next level sweeping the search across the neighour nodes at that level and so on until the objective is reached. When all the nodes are searched, but the objective is not met then it is reported as failure.

Breadth first is well suited for situations where the objective is found on the shallower parts in a deeper tree. It will not perform so well when the branches are so many in a game tree, especially like chess game and also when all the path leads to the same objective with the same length of the path [7] [8].

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 optimizations.

3.2 Depth First Search Algorithm

This powerful technique of systematically traverse through the search by starting at the root node and traverse deeper through the child node. If there are 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 [10].

This algorithm makes sure that all the edges are visited once breadth [11]. It is well suited for search problems, but when the branches are large then this algorithm takes might end up in an infinite loop [8].

3.3 Page Rank Algorithm

Page rank algorithm determines the importance of the web pages by counting citations or backlinks to a given page [12]. 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 Website,

d → damping factor

$T_1, \ldots T_n$ → links

Yongbin Qin and Daoyun Xu [13] proposed an algorithm, taking the human factor into consideration, to introduce page belief recommendation mechanism and brought forward a balanced rank algorithm based on PageRank and page belief recommendation which ultimately attaches importance into the subjective needs of the users; so that it can effectively avoid topic drift problems. Tian Chong [14] 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 [15] proposed a dynamic page ranking algorithm. Shaojie Qiao [16] 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)

3.4 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 exists but the technique is to find the best solution within specified time [17].

[18] shows the genetic algorithm is best suited when the user has literally no or less time to spend in searching a huge database and also very efficient in multimedia results. While almost all conventional methods search from a single point, Genetic Algorithms always operates on a whole population. This contributes much to the robustness of genetic algorithms. It reduces the risk of becoming trapped in a local stationary point [19]. The applicability of Genetic Algorithms by various researchers [20],[21], [22], [23], [24], [25], [26] has been depicted in [27].

3.5 Naïve Bayes classification Algorithm

Naïve Bayes algorithm is based on Probabilistic learning and classification. It assumes that one feature is independent of another [28]. This algorithm proved to be efficient over many other approaches [29] although its simple assumption is not much applicable in realistic cases [28].

Wenxian Wang et al [30] proposed an efficient crawler based on Naïve Bayes to gather many relevant pages for hierarchical website layouts. Peter Flach and Nicolas Lachiche [31] presented

Naïve Bayes classification of structured data on artificially generated data.

3.6 HITS Algorithm

This algorithm put forward by Kleinberg is previous to Page rank algorithms which uses scores to calculate the relevance [32]. 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 [2].

Joel C. Miller et al [33] proposed a modification on adjacency matrix input to HITS algorithm which gave intuitive results.

4. 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 favors more for Genetic Algorithm due to its iterative selection from the population to produce relevant results .

References:

[1] Pavalam S M, Jawahar M, Felix K Akorli, S V Kashmir Raja " Web Crawler in Mobile Systems" International Conference on Machine Learning (ICMLC 2011), Vol. , pp

[2] Alessio Signorini, "A Survey of Ranking Algorithms" retrieved from http://www.divms.uiowa.edu/~asignori/phd/report/a-survey-of-ranking-algorithms.pdf 29/9/2011

[3] Maurice de kunder, "Size of the world wide web", retrieved from http://www.worldwidewebsize.com/ 8/8/11

[4] Ricardo Baeza-Yates, Ricardo Baeza-Yates "Crawling a Country: Better Strategies than Breadth-First for Web Page Ordering" , Proc. WWW 2005.

[5] Marc Najork, "Web Crawler Architecture" retrieved from http://research.microsoft.com/pubs/102936/EDS-WebCrawlerArchitecture.pdf accessed on 10/8/11

[6] Junghoo Cho and Hector Garcia-Molina "Effective Page Refresh Policies for Web Crawlers" ACM Transactions on Database Systems, 2003.

[7] Steven S. Skiena "The Algorithm design Manual" Second Edition, Springer Verlag London Limited, 2008, Pg 162.

[8] Ben Coppin "Artificial Intelligence illuminated" Jones and Barlett Publishers, 2004, Pg 77.

[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] Alexander Shen "Algorithms and Programming: Problems and solutions" Second edition Springer 2010, Pg 135

[11] Narasingh Deo "Graph theory with applications to engineering and computer science" PHI, 2004 Pg 301

[12] Sergey Brin and Lawrence Page "Anatomy of a Large scale Hypertextual Web Search Engine" Proc. WWW conference 2004

[13] Yongbin Qin and Daoyun Xu "A Balanced Rank Algorithm Based on PageRank and Page Belief recommendation"

[14] 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)

[15]J.Kleinberg "Authoritative sources in a hyperlinked environment", Proc 9th ACM-SIAM Symposium on Discrete Algorithms, 1998.

[16] Shaojie Qiao, Tianrui Li, Hong Li and Yan Zhu, Jing Peng, Jiangtao Qiu "SimRank: A Page Rank Approach based on similarity measure" 2010 IEEE

[17] S. N. Sivanandam, S. N. Deepa "Introduction to Genetic Algorithms" Springer, 2008, pg 20

[18] 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

[19] Deep Malya Mukhopadhyay, Maricel O. Balitanas, Alisherov Farkhod A.,Seung-Hwan Jeon, and Debnath Bhattacharyya "Genetic Algorithm: A Tutorial Review" International Journal of of Grid and Distributed Computing Vol.2, No.3, September, 2009.

[20] Shian-Hua Lin, Jan-Ming Ho, Yueh-Ming Huang ,ACRID ,intelligent internet document organization and retrieval ,IEEE Transactions on Knowledge and data engineering, 14(3),559-613, 2002

[21] D.H. Kraft, F.E. Petry, B.P. Buckes, T. Sadasivan, Genetic algorithm for query optimization in information retrieval: relevance feedback, in: E. Sanchez, T. Shibata, L.A. Zadeh (Eds.), Genetic Algorithms and Fuzzy Logic Systems, 1997, pp. 155–173.

[22] E. Sanchez, H. Miyano, J. Brachet, Optimization of fuzzy queries with genetic algorithms. Applications to a database of patents in biomedical engineering, in: Proc. VI IFSA Congress, Sao- Paulo, Brazil, 1995, pp. 293–296.

[23] Zacharis Z. Nick and Panayiotopoulos Themis, Web Search Using a Genetic Algorithm, IEEE Internet computing,1089-7801/01c2001, 18-25, IEEE

[24] Ramakrishna Varadarajan, Vagelis Hristidis, and Tao Li , Beyond Single-PageWeb Search Results,IEEE Transactions on knowledge and data engineering, 20(3) ,411 - 424, 2008

[25] Judit BarIlan, Comparing rankings of search results on the Web, Information Processing and Management 41 (2005) 1511–1519

[26] Adriano Veloso, Humberto M. Almeida, Marcos Goncalves, Wagner Meira Jr.,Learning to Rank at Query-Time using Association Rules, SIGIR'08, 267-273 , 2008, Singapore.

[27] S.Siva Sathya and Philomina Simon," Review on Applicability of Genetic Algorithm to Web Search" International Journal of Computer Theory and Engineering, Vol. 1, No. 4, October2009

[28] Harry Zhang "The Optimality of Naive Bayes" American Association for Artificial Intelligence 2004.

[29] Rich Caruana, Alexandru Niculescu-Mizil "An Empirical Comparison of Supervised Learning Algorithms" Proc 23 rd International Conference on Machine Learning, Pittsburgh, PA, 2006.

[30] 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

[31] Peter A. Flach and Nicolas Lachiche "Naive Bayesian Classification of Structured Data" Machine Learning, Kluwer Academic Publishers

[32] Kleinberg, John "Hubs, Authorities, and Communities" ACM computing survey,1998.

[33] 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.