Differences in Caching of Robots.txt by Search Engine Crawlers

Jeeva Jose

Department of Computer Applications
Baselios Poulose II Catholicos College, Piravom
Ernakulam District, Kerala, India
vijojeeva@yahoo.co.in

Abstract

Web Log Mining gives insight to the behavior of search engine crawlers accessing a Website. Crawlers periodically visit the Website and update contents on the Website. The behavior of search engine crawlers gives vital information about the ethics of crawlers, dynamicity of crawling, how much they contribute to the server load and so on. Ethical crawlers initially access the "robots.txt" file and then proceeds to the crawling process according to the permissions and restrictions given in this file. This paper is an attempt to identify the differences of various search engine crawlers and the time delay in caching the "robots.txt" file. The results revealed that there is a significant difference in the caching of "robots.txt" file by various crawlers.

Keywords- Web Log Mining; Search Engine Crawlers; robots.txt; cache

I. INTRODUCTION

Search engine crawlers are automated programs which periodically visit a Website to update information. Crawlers are also known as 'bots', 'spiders', or 'robots'. Crawlers are the main components of a search engine and without them the Websites will not be listed in search results. The visibility of the Websites depends on the quality of the crawlers. Search engines such as Google periodically use Web robots to grab all the pages from a Website to update their search indexes. If the site doesn't attract many visitors, the number of requests from all the Web robots that have visited the site might exceed that of human generated requests [1].

Search engines do not index sites equally, may not index new pages for months, and no engine indexes more than about 16% of the Web [2]. Certain crawlers avoid too much load on a server by crawling the server at a low speed during peak hours of the day and at a high speed during late night and early morning. A crawler for a large search engine has to address two issues.

1)It has to have a good crawling strategy which means a strategy for deciding which pages to download next.

2)It needs to have a highly optimized system architecture that can download a large number of pages per second while being robust against crashes, manageable of resources and Web servers [3].

The Web creates new challenges for information retrieval. The amount of information on the Web is growing rapidly as well as the number of new users inexperienced in the art of Web search. There are several works in literature which have studied about the design and behavior of search engine crawlers. Reference [4] provides a detailed anatomy of the large hyper textual Web search engine, Google. It explains the design goals, system features, scalability, improved search quality, the PageRank calculation, the detailed overview of its architecture, data structures, performance and comparison with other crawlers. PageRank is the backbone of Google.

Web Mining tasks include mining Web search engine data, analyzing Web's link structures, classifying Web documents automatically, mining Web page semantic structures and page contents, mining Web dynamics (mining log files), building a multilayered and multidimensional Web. Web log data is usually mined to study the user behavior at Websites. It also contains immense information about the search engine traffic contributed by crawlers. The user traffic is removed by pre-processing tasks, otherwise it may bias the search engine behavior. Thus the refined data enables to analyze search engine crawler behavior. The search engine crawler is an important module of a Web search engine and the quality of a crawler directly affects the searching quality of Web search engines.

The process of identifying the Web crawlers is important because they can generate 90% of the traffic on Websites [5]. Commercial search engines play a vital role in accessing Websites and wider information dissemination [6][7]. A typical crawler starts with a seed set of pages. It then downloads these pages, extracts hyperlinks and crawls pages pointed to by these new hyperlinks. The crawler repeats this step until there are no more pages to crawl or some resources (e.g. time or network bandwidth) are exhausted [8]. These crawlers are highly automated and seldom regulated manually.

The crawlers periodically visit the Websites to update the content. Certain Websites like stock market sites or online news may need frequent crawling to update the search engine repositories. Web crawlers access the Websites for diverse purpose which includes security violations also. Hence they may lead to ethical issues like

privacy, security and blocking of server access. Crawling activities can be regulated from the server side with the help of Robots Exclusion Protocol [9]. This protocol is present in a file called robots.txt. Robots.txt is a file that Web agents (crawlers) check for information on how the site is to be catalogued. It is a text file that defines what documents and/or directories are forbidden which are followed by ethical crawlers. There is also an option to direct the Web agents (crawlers) per page.

Every HTML document contains a heading section in which meta-data about the document (like keywords, a description of the content, and so on) can be included. Such sections are called 'meta tags.' Within the meta-tags of each HTML document one can specify whether or not a robot is allowed to index the page and submit it to a search engine. Some robots will simply ignore the meta- tags because of the fact that those tags are often misused by page owners who want to get a higher ranking in a search index.

Robots (crawlers) may also ignore the robots.txt file or purposely load the documents that the file marks as disallowed. But it is also possible to crawl the pages at a Website without accessing the robots.txt. Certain crawlers seems to disobey the rules in robots.txt after its modification because crawlers like "Googlebot", "Slurp" , "MSNbot" cache the robots.txt file for a Website [10]. Usually ethical crawlers first access this file which will be present at the root directory of the Website and follow the rules specified by robots.txt [11]. Certain pages and folders are denied access because they contain sensitive information which is not intended to be publically available. There may be situations where two or more versions of a page will be available one as html and other one as pdf. The crawlers can be made to avoid crawling the pdf version for eliminating redundant crawling. Also certain files like JavaScripts, images, stylesheets etc. can be avoided for saving the time and bandwidth. The structure of a robots.txt file is follows.

User-agent:

Disallow:

"User-agent:" is the search engine crawler and "Disallow:" lists the files and directories to be excluded from indexing. In addition to "User-agent:" and "Disallow:" entries, comment lines are included by putting the # sign at the beginning of the line. For example all user agents are disallowed from accessing the folder /a. This is given in robots.txt as follows.

All user agents are disallowed to see the /a folder.

User-agent: *

Disallow: /a/

The Website monitoring software Google Analytics does not track crawlers or bots. This is because Google Analytics tracking is activated by a JavaScript that is placed on every page of the Website. A crawler hardly recognizes these scripts and hence the visits from search engines are not recognized.

II. BACKGROUND LITERATURE

Search engines largely rely on Web crawlers to collect information from the Web. Due to the unregulated open-access nature of the Web, crawler activities are extremely diverse. Such crawling activities can be regulated from the server side by deploying the Robots Exclusion Protocol in a file called robots.txt. The study of robots.txt is done in [9][10]. Current day crawlers retrieve content only from the publicly indexable Web. This includes the set of Web pages reachable purely by following hypertext links, ignoring search forms and pages that require authorization or prior registration. In particular, they ignore the tremendous amount of high quality content "hidden" behind search forms, in large searchable electronic databases.

Reference [12] has provided a framework for addressing the problem of extracting content from this hidden Web. Scalable Web crawlers are an important component of many Web services. Building a scalable crawler is a non-trivial endeavor because the data manipulated by the crawler is too big to fit entirely in memory, so there are performance issues relating to how to balance the use of disk and memory. Reference [13] has enumerated the main components required in any scalable crawler and it has discussed design alternatives for those components. In particular, it has described Mercator, an extensible scalable crawler written entirely in Java. Mercator's design features a crawler core for handling the main crawling tasks. Extensibility is achieved through protocol and processing modules.

The traditional information retrieval measures of recall and precision at varying numbers of retrieved documents were calculated and used these as the basis for statistical comparisons of retrieval effectiveness among the eight search engines [14]. A study on major search engines and directories and cites why these search engines are the toppers in the list is conducted in [15].

Reference [8] has proposed the need for Web servers to export meta data describing their pages so that crawlers can efficiently create and maintain large, fresh repositories. This meta-data includes the last modified date and size for each available file which if exported could save considerable amount of band width. Reference [16] reports on an experiment to investigate the effect of link count on the indexing of 1000 sites in three search portals over a period of seven months. It was found that, although all search engines added sites during the period

of the survey, only Google showed evidence of being very responsive to the existence of links on the test site, whereas AltaVista's results were very stable over time. Due to limited bandwidth, computational resources and dynamic nature of the Web, search engines cannot index every Web page and even the covered pages cannot be monitored continuously for changes. It is important to develop crawling strategies to prioritize the pages to be indexed. For topic specific search engine crawlers, this is more important.

III. PRE-PROCESSING

In this work the log file of a business organization NeST was selected for study. The dataset ranges from May 1, 2014 to May 31, 2014. The log file is extracted and data pre-processing is done to eliminate the user requests since the focus is on the behavior of search engines. After extracting the data set, it is found that the extracted data consists of 5,29,175 records. The entries with unsuccessful status code are eliminated.

The HTTP requests with POST and HEAD is also removed. In addition, all the user requests are removed to get the search engine requests. This is required as a user request in the input file may bias the results of search engine crawler behavior. After pre-processing the resultant file contained only the successful search engine crawler requests. Reference [17] uses three heuristics to identify robots (crawlers).

1) Look for all hosts that have requested the page "robots.txt."
2) Use a list of user agents known as robots.
3) Compute the browsing speed. Browsing speed is computed as (Number of viewed pages)/(session time). If browsing speed exceeds a threshold θ1 (pages/second), and the number of visited pages for that visit exceeds a threshold θ2, it is considered to be a Web robot.

In this work some crawlers are identified from the IP address field. It contained substrings like "Googlebot", "Baiduspider", "MSNbot" etc. The user agents are also helpful in identifying the bots or crawlers like Ezooms, Discobot etc. The referrer URL containing "robots.txt" is also considered as the request from search engine crawlers. Various search engine crawlers are identified from our data set. Certain search engine crawlers with number of visits less than 5 per week is removed as it is considered irrelevant. There were several unethical crawlers which didn't access the robots.txt file. Four ethical crawlers Bingbot, Googlebot, MSNbot and slurp were considered for analysis.

IV. DIFFERENCES IN DELAY OF CACHING THE ROBOTS.TXT

Bingbot is the crawler for Bing search engine. It was developed by Microsoft. Googlebot is a Web crawling spider from Google. Googlebot uses huge set of computers to crawl billions of pages on the Web. It uses an algorithmic process which involves computer programs to determine which sites to crawl, how often, and how many pages to fetch from each site. Googlebot's crawl process begins with a list of Webpage URLs, generated from previous crawl processes and augmented with sitemap data provided by Webmasters. As Googlebot visits each of these Websites it detects links SRC and HREF on each page and adds them to its list of pages to crawl. New sites, changes to existing sites and dead links are noted. This is used to update the Google index. Usually on an average Googlebot access the site not more than once every few seconds. However, due to network delays, it is possible that the rate will appear to be slightly higher over short periods. In general, Googlebot download only one copy of each page at a time. If Googlebot is downloading a page multiple times, it is probably because the crawler was stopped and restarted. Googlebot was designed to be distributed on several machines to improve performance and scale as the Web grows. To reduce the bandwidth usage, many crawlers on machines located near the sites are sent. Therefore, the logs may show visits from several machines at google.com, all with the user-agent Googlebot.

MSNbot is a crawler developed by Microsoft for MSN search engine. MSN search engine offers Webmasters the ability to slow down the crawl rate to accommodate Web server load issues. Websites that are small in terms of the number of pages and whose content is not regularly updated probably will never need to set crawl delay settings. The bot will automatically adjust its crawl rate to an appropriate level based on the content it finds with each pass. Larger sites that have a great many pages of content may need to be crawled more deeply and more often so that their latest content may be added into the index. Slurp is the Web crawler from Yahoo. The user agent for slurp is Mozilla/5.0 (compatible; Yahoo! Slurp;). The original developer of Slurp was Inktomi and later Yahoo acquired Inktomi. Table I shows the four crawlers with the number of times the robots.txt was cached during one month.

TABLE.I CRAWLERS WITH NUMBER OF CACHES FOR ROBOTS.TXT

Search Engine Crawler	No: of Caching of Robots.txt
Bingbot	607
Googlebot	89
MSNbot	35
Slurp	74

A) Analysis of Variance (ANOVA)

Analysis of Variance (ANOVA) is used to analyze the variations in the collection of data among groups and within groups [18]. ANOVA conducted using a single factor is known as one way ANOVA. In this data set time delay in seconds for caching the robots.txt of 4 search engine crawlers are chosen for study. The single factor considered is the time elapsed between two consecutive caches of robots.txt by search engine crawlers. The following null hypothesis H_0 and alternate hypothesis H_1 is considered for ANOVA.

H_0: The means are equal.

H_1: The means are not equal.

Let k: the number of levels or groups in the experiment, N: total number of subjects in the experiment, n: number of subjects in each group, T: ∑X for each group, G: ∑X for the entire experiment. Table II gives the formula summary for ANOVA.

TABLE II FORMULA SUMMARY FOR ANOVA

Source	df	SS	MS	F	p
Between Groups	k-1	$\sum \frac{T^2}{n} - \frac{G^2}{N}$	$\frac{SS_{BG}}{df_{BG}}$	$\frac{MS_{BG}}{MS_{WG}}$	If P > 0.10 No evidence against the null hypothesis. If 0.05 < P < 0.10, Weak evidence against the null hypothesis. If 0.01 < P < 0.05 Moderate evidence against the null hypothesis. If 0.001 < P < 0.01 Strong evidence against the null hypothesis. If P < 0.001 Very strong evidence against the null hypothesis .
Within Groups	N-k	$\sum$SSinside each group	$\frac{SS_{WG}}{df_{WG}}$		
Total	N-1	$\sum X^2 - \frac{G^2}{N}$			

The detailed statistic descriptive of the time delay in second between the caching of robots.txt by various search engine crawlers is given in Table III and results of One Way ANOVA is given in Table IV.

TABLE III STATISTIC DESCRIPTIVES

Cache Delay of Robots.txt

Search Engine Crawlers	N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean		Minimum	Maximum
					Lower Bound	Upper Bound		
Bingbot	607	4275.42	3517.198	142.759	3995.06	4555.78	0	20880
Googlebot	89	27216.40	22761.498	2412.714	22421.64	32011.17	840	86940
MSNbot	35	60553.71	102903.256	17393.825	25205.21	95902.22	360	367260
Slurp	77	31106.49	14815.510	1688.384	27743.79	34469.20	1080	57240
Total	808	11797.05	27203.679	957.022	9918.51	13675.60	0	367260

TABLE IV RESULTS OF ONE WAY ANOVA

Cache Delay of Robots.txt

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	167413537901.901	3	55804512633.967	104.390	.000
Within Groups	429798853087.703	804	534575687.920		
Total	597212390989.604	807			

The significance is .000 and we reject the null hypothesis. Hence we conclude that there is a significant difference in the mean time delay between the caching of robots.txt file by various search engine crawlers.

B) Duncan's Multiple Range Test

Post hoc tests are used for situations in which the results have already obtained a significant omnibus F-test with a factor that consists of three or more means and additional exploration of the differences among means is needed to provide specific information on which means are significantly different from each other [19]. Duncan's Multiple Range Test is a post hoc test. Table V shows the results of Duncan's Multiple Range Test. The significant difference or the range value is given by

$$R_p = r_{\alpha,p,v}\sqrt{MSE/n}$$

where rα,p,v is the Duncan's Significant Range Value with parameters alpha level α = αjoint, p= range value and v = MSE degrees of freedom. MSE is the mean square error from the ANOVA table and n is the number of observations used to calculate the means being compared.

TABLE V DUNCAN"S MULTIPLE RANGE TEST

Search Engine	N	Subset for alpha = 0.05		
		1	2	3
Bingbot	607	4275.42		
Googlebot	89		27216.40	
MSNbot	77		31106.49	
Slurp	35			60553.71
Sig.		1.000	.308	1.000

V. CONCLUSION

The results revealed that Bingbot cached more number of times compared to Googlebot, MSNbot and Slurp. ANOVA revealed that there is a significant delay in caching the robots.txt file by various crawlers. Also there is a significant time delay between caching of robots.txt among the same crawler also. The Duncan's Multiple Range Test revealed that the caching behavior of Googlebot and MSNbot was very close while Bingbot and Slurp behaved in a different way. The more the number of caches, the more a search engine crawler is likely to obey the Robot Exclusion protocol.

ACKNOWLEDGMENT

This research work is funded by University Grants Commission, New Delhi, as per order No.1998-MRP/14-15/KLMG066/UGC-SWRO dated 04-Feb-15.

REFERENCES

[1] D. Tanasa and B. Trousse, "Advanced Data Preprocessing for Intersites Web Usage Mining", Intelligent Systems, vol. 19, no. 2, pp.59-65, 2004.
[2] S. Lawrence and C. L. Giles, "Accessibility of Information on the Web", Nature, vol. 400, no. 107, pp. 107-109, 1999.
[3] V. Shkapenyuk and T. Suel, "Design and Implementation of a High-Performance Distributed Web Crawler", in Proc. of the 18th Int. Conf. on Data Eng., Washington, DC, USA, pp. 357-368, 2002.
[4] S.Brin and L. Page, "The Anatomy of a Large Scale Hypertextual Web Search Engine", Comput. Networks and ISDN Syst. vol.30, pp. 107- 117, 1998.
[5] D. Mican and D. Sitar-Taut," Preprocessing and Content/ Navigational Pages Identification as Premises for an Extended Web Usage Mining Model Development", Informatica Economica,vol. 13, no.4, pp.168-179, 2009.
[6] D. Sullivan, Webspin. (2003). Newslett. [online]. Available http://contentmarketingpedia.com/Marketing-Library/Search/ industry NewsSeptA1.pdf, Retrieved December 4, 2012.
[7] L. Vaughan and M. Thelwall, "Search Engine Coverage Bias: Evidence and Possible Causes", Inform. Process. and Manage., vol. 40, no.4, pp. 693-707, 2004.
[8] O. Brandman, "Crawler-Friendly Web Servers",SIGMETRICS Newslett., vol. 28, no.2, pp.9-14, 2000.
[9] Y. Sun et al., "A Large-Scale Study of Robots.txt", in Proc. of the 16th Int. WWW conf., Banff, Canada, 2007, 1123-1124.
[10] M. Drott, "Indexing aids at corporate Websites: The use of robots.txt and meta tags", Inform. Processing and Manage., vol.38, no.2, pp. 209-219, 2002.
[11] L. Giles et al.,"Measuring the Web Crawler Ethics", in Proc. of the 19th Int.WWW Conf., Raleigh, USA, 2010, pp. 1101-1102.
[12] S. Raghavan and H. Garcia-Molina, "Crawling the Hidden Web", Tech. Repo.,2000-36, Computer Science Department, Stanford University, December 2000. Available at http://dbpubs.stanford.edu/pubs/2000-36
[13] A. Heydon and M. Najork, "Mercator: A scalable, extensible Web crawler", World Wide Web, vol.2, no. 4, pp.219-229, 1999.
[14] F. M. Gordon and P. Pathak, "Finding information on the World Wide Web: the retrieval effectiveness of search engines", Inform. Processing and Manage., vol. 35, pp. 141-80, 1999.

[15] D. Sullivan, "Major Search Engines and Directories", [online], Available: http://www.leepublicschools.net/Technology/Search-Engines_Directories.pdf, Retrieved on May 15, 2013.
[16] M. Thelwall, "The Responsiveness of Search Engine Indexes", Int. J. of Scientometrics, Informetrics and Bibliometrics, vol. 5, no. 1, pp. 1-10, 2001.
[17] D. Tanasa and B. Trousse, "Advanced Data Preprocessing for Intersites Web Usage Mining", Intelligent Systems, vol. 19, no. 2, pp.59-65, 2004.
[18] R. Paneerselvam, Research Methodology, Prentice Hall of India, 2nd ed., 2004.
[19] D. B. Duncan, "Multiple range and multiple F tests", Biometrics II, pp.1-42, 1955.