FOCUS: ADAPTING TO CRAWL INTERNET FORUMS

T.K. Arunprasath, Dr. C. Kumar Charlie Paul

Abstract— **Internet is emergent exponentially and has become progressively more. Now, it is complicated to retrieve relevant information from internet. The rapid growth of the internet poses unprecedented scaling challenges for general purpose crawlers and search engines. In this paper, we present a novel Forum Crawler under Supervision (FoCUS) method, which supervised internet-scale forum crawler. The intention of FoCUS is to crawl relevant forum information from the internet with minimal overhead, this crawler is to selectively seek out pages that are pertinent to a predefined set of topics, rather than collecting and indexing all accessible web documents to be capable to answer all possible ad-hoc questions. FoCUS is continuously keeps on crawling the internet and finds any new internet pages that have been added to the internet, pages that have been removed from the internet. Due to growing and vibrant activity of the internet; it has become more challengeable to navigate all URLs in the web documents and to handle these URLs. We will take one seed URL as input and search with a keyword, the searching result is based on keyword and it will fetch the internet pages where it will find that keyword.**

Index Terms— **EIT path, forum crawling, ITF regex, page classification, page type, URL pattern learning, URL type**

I. INTRODUCTION

The advent of the Internet has become more powerful to get source of information. It has been widely used by a user who belongs to a variety of avenues. The need of the hour is to make the process of searching the internet for information more and more efficient. With the size of the internet growing exponentially, the volume of data to be crawled also p roportionally grows, as a result of which it becomes progressively more necessary to have appropriate crawling mechanisms in order to make crawls efficient. This has made engineering a search a more challengeable work. FOCUS basically performs the three basic tasks namely: They search the Internet or select pages on important words. They keep an index of the words they stumble on and where they stumble on them. They allow to users to see the words or combination of words found in that index. A WebCrawler is a computer program that browses the Internet in a methodological manner. The crawler typically crawls through links grabbing information from websites and adding it to search engines indexes. Internet provides a vast source of information of almost all type. But, this information is often scattered among many web servers and hosts, using many different formats. We all want that we should have the most excellent promising search manner in less time.

In this paper, we present Forum Crawler under Supervision (FoCUS), supervised web-scale forum crawler, which is merged, with the progression for finding the copyright infraction. For any crawler there are two issues: First, the crawler should have the capability to plan, i.e., whose plan will settle on which pages are going to download next. Second, it needs to have exceedingly optimized and vigorous system architecture so that it can download an number of pages per second even next to crashes, manageable, and considerate of resources and web servers. Some recent intellectual interest is there in the first issue, including work on deciding which pages crawler should obtain first. In contrast, few work is done on second issues. Clearly, all the major search engines have vastly optimized crawling system, although working and particulars of documentation of this system are usually with their owner. It is easy to construct a crawler which would work slowly and download few pages per second for a petite period of time. In contrast, it's a big confront to build the same system design, I/O, network efficiency, robustness and manageability. Every search engine is alienated into different modules among those modules; crawler module is one of the modules on search engine which relies on the majority because it helps to afford the best probable results to FOCUS. FOCUS is small techniques which 'browse' the web on the search engine's behalf, similarly to come across "how a human user would follow links to reach different pages". The programs are given a starting seed URLs, whose pages they retrieve from the web. The crawler extracts URLs appearing in the retrieved pages, and gives this information to crawler for control the module. This module determines what links to visit next, and feeds the links to visit back to the crawlers. The crawler also passes the retrieved pages into a page repository. Crawlers continue visiting the web, until local resources, such as storage, are fatigued. Our contribution of work follows as:

1. Identify the bad URL in the website.
2. Identify type of protocol used for any web page.
3. Retrieve the web pages, we apply pattern recognition over text and pattern symbolizes check text only.
4. Check how much text is available on web page.

The rest of the method proceeds as follows: In Section I, we formally introduce the system model and the ideas of Focused crawler method for retrieve the exact information form internet. In Section II, we discussed about related work for understanding the previous work. In section III, we presented proposed scheme and their framework for implementation of FoCUS.

Manuscript received Dec, 2013.

T.K.Arunprasath*, Computer Science and Engineering, A.S.L Pauls college of engineering and technology, Coimbatore, India.*

Dr. C. Kumar Charlie Paul*, Computer Science and Engineering, A.S.L Pauls College of Engineering and Technology, Coimbatore, India,.*

ISSN: 2278 – 7798

II. Related work

Vidal et al. [25] proposed a tale approach for learning regular expression patterns of URLs that lead a crawler from an entry page to target pages.Target pages were found through comparing the DOM trees of pages with a preselected trial of target page. It is very effective but it only works for the particular site from which the sample page is drawn.It essential to repeat the same process for every time for the new site. However, it is not pertinent for large-scale crawling. In contrast, our proposed approach FoCUS which learns URL patterns across multiple sites and automatically finds a forum's entry page given a page from the forum. Guo et al. [17] and Li et al. [20] are comparable to our work. However, they did not mention "how to discover and traverse the URLs". Li et al. developed some heuristic rules to discover URLs. But, the rules are very specific and it can only be applied to specific forums powered by the particular software package in which the heuristics were conceived. Unfortunately, according to the Forum Matrix [2], there is lot of discomparable forum software packages used on the Internet.

Wang et al. [26] presented an algorithm to address the traverse path selection problem. They introduced the scheme of skeleton link and page-flipping link. Skeleton links are "the most significant links supporting the structure of a forum site." Importance is determined by the informativeness and coverage metrics. Page-flipping links are determined using the connectivity metric. By identifying and only following skeleton links and page-flipping links, they demonstrated that iRobot can achieve effectiveness and coverage. According to our supervision, the sampling strategy and informativeness estimation is not stout and tree-like traversal path is not possible. Traversal path does not tolerate more than one path from a starting page node to a same ending page node.

Another related work is in the vicinity of our work which presented to avoid duplicate detection. Forum crawling also desires to remove duplicates. However, this content based duplicate detection [18], [21] does not have competent bandwidth, it can only be carried out when pages have been downloaded. URL-based duplicate detection [14], [19] is not supportive. In forums, index URLs, thread URLs, and page-flipping URLs have specific URL patterns. Thus, in our paper, by learning patterns of index URLs, thread URLs, and page-flipping URLs and adopting a simple URL string de-duplication technique (e.g., a string hashset), FoCUS can be easily avoided duplicates without any duplicate detection. To advance the unnecessary crawling, industry standards such as "no follow" [6], Robots Exclusion Standard (robots. txt) [10], and Sitemap Protocol [9], [22] have been introduced here. By specifying the "rel" attribute with the "no follow" value (i.e., "rel ¼ nofollow"), page authors can inform a crawler that the destination content is not endorsed. However, it is intended to diminish the effectiveness of search engine spam, but not meant for blocking the access to pages. A proper way is robots.txt [10]. It is designed to identify what pages a crawler is allowed to visit or not. Sitemap [9] is an XML file that lists the URLs along with additional metadata including update time, change frequency and efficiency etc. Generally , the intention of robots.txt and Sitemap is to facilitate to be crawled intelligently. So they may be useful to forum crawling. However, it is complicated to sustain such files for forums as their content continually changes.

III. Proposed Scheme

In this we discussed about our proposed scheme and how to implement it. In this section, we illustrate all the method in separate module with detailed description such as synopsis of anticipated scheme, ITF Regexes Learning, Online Crawling and Entry URL Discovery.

A. Overview Proposed Scheme

In this section we present architectural diagram for our anticipated scheme in Fig.1. It consists of two major parts: the learning part and the online crawling part. The learning part first learns ITF regexes of a given forum from automatically constructed URL training examples. The online crawling part then applies learned ITF regexes to crawl all threads efficiently.

1. Page Type: In this module, **w**e classified the forum pages into following page types.

Entry Page: The homepage of a forum, which contains a list of boards and is also the lowest familiar ancestor of all threads.

Index Page: A page of a board in a forum, which typically contains a table-like structure and which contains information of a board or a thread. The list-of board page, list-of-board and the thread page, and the board page are all index pages.

Thread Page: A page of a thread in a forum that contains a list of posts with user generated content belonging to the comparable discussion.

2. URL Type: In this module, we discuss about types of URL

Index URL: A URL that is on an entry page or index page and points to an index page. Its anchor text shows the title of its destination board.

Thread URL: A URL that is on an index page and points to the thread page. Its anchor text is title of the destination thread

B. *ITF Regexes Learning*

In this section, we learn about ITF regexes, FoCUS which adopts two-step supervised training procedure. The first step is training sets construction. The second step is regexes learning.

1. Constructing URL Training Sets:

The goal of URL training sets construction is to automatically construct the sets of highly precise index URL, thread URL, and page-flipping URL strings for ITF regexes learning. We use a comparable process to construct index URL and thread URL training sets since they have very comparable properties with the exception of the types of their destination pages.

2. Learning ITF Regexes:

In this sub-module, **w**e have shown how to construct index URL, thread URL, and page-flipping URL string training set. We also elucidate how to learn ITF regexes from these training sets. Vidal et al. [25] applied URL string generalization. For example, given URLs as follows (the top four URLs are encouraging while the bottom two URLs are pessimistic):

http://www.gardenstew.com/about20152.html

ISSN: 2278 – 7798

http://www.gardenstew.com/about18382.html
http://www.gardenstew.com/about19741.html
http://www.gardenstew.com/about20142.html
http://www.gardenstew.com/user-34.html
http://www.gardenstew.com/post-180803.html
It creates a URL regular expression pattern as follows: http://www.gardenstew.com/\w+\W+\d+.html; while the target pattern is http://www.gardenstew.com/ about\d+.html. Instead, we apply the method introduced by Koppula et al. [19] which is advanced to deal with pessimistic examples.

C. *Online Crawling*

In this module, we perform online crawling using a breadth-first strategy (actually, it is easy to adopt other strategies). FoCUS first pushes the entry URL into a URL queue; next it fetches a URL from the URL queue and finally downloads its page; and then it pushes the outgoing URLs which are coordinated with any learned regex into the URL queue. FoCUS repeats this step until the URL queue is empty or other conditions are satisfied. FoCUS only needs to apply the learned ITF regexes on innovative outgoing URLs in newly downloaded pages to making the more proficient for online crawling. FoCUS does not need to group outgoing URLs, classify pages, recognize page-flipping URLs, or learn regexes again for that forum.

D.Entry URL Discovery

In this module, an entry URL needs to be precise to start the crawling process. In particular in web-scale crawling, manual forum entry URL bad notation is not practical. Forum entry URL discovery is not a trivial task since entry URLs vary from forums to forums. We developed a novel heuristic rule to stumble on entry URL as a baseline. The heuristic baseline tries to stumble on the following keywords ending with "/" in a URL: forum, board, community, bbs, and discus. If a keyword is found, the path from the URL host to this keyword is extracted as its entry URL; if not, the URL host is extracted as its entry URL. To make the FoCUS more practical and scalable, we design a simple yet effective forum entry URL discovery method based on some techniques.

IV. RESULT

In this section, we show the result which are we proposed implement in previous section. In this section, we show all the result with the help of table and graph in separate module with detailed description such as overview, Online Crawling and Entry URL Discovery.

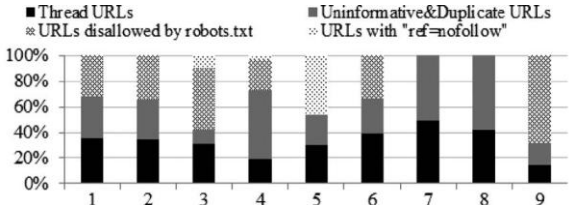
A. Entry URL Discovery

In this module, we discuss, forum crawling assume in URL Entry. However, finding forum entry URL is not trivial. To display this, we used our URL entry discovery method with a heuristic baseline. For each forum in the test set, we randomly sampled a page and fed it to this module. Then, we checked manually if the output was indeed its entry page. In order to see whether FoCUS and the baseline were robust or not, we repeated this process 10 times with unusual sample pages. The results are shown in Table 1. The baseline had 76 percent precision and recall. On the contrary, FoCUS achieved 99 percent precision and 99 percent recall. The low standard deviation also designates that it is not sensitive to sample pages. There are two main failure cases: 1) forums are no longer in operation and 2) JavaScript generated URLs which we do not handle currently. We balanced the different types of URL for find the efficiency of thread URL and URL discovery in terms of generic crawler in figure-02

TABLE 1- Results of Entry URL Discovery

Method	Precision %		Recall %	
	Average	Std. Dev.	Average	Std. Dev.
Baseline	76.38	1.74	76.38	1.74
FoCUS	99.31	0.20	99.13	0.32

Fig.- 1. Ratio of different URLs discovered by a generic crawler



B. Evaluation of Online Crawling

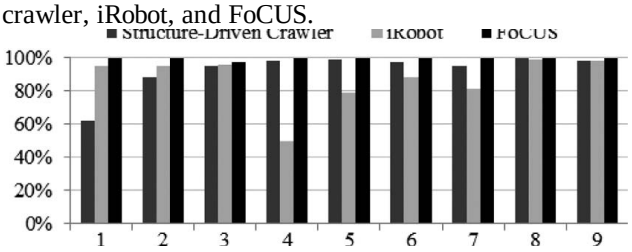
In this module, we evaluate FoCUS with other existing methods for find the efficiency of result

TABLE 2- Forums Used in Online Crawling Evaluation

ID	Forum	Forum Name	Software	#Threads
1	forums.afterdawn.com	AfterDawn: Forums	Customized	535,383
2	forums.asp.net	ASP.NET Forums	Community Server	66,966
3	forum.xda-developers.com	Android Forums	vBulletin	299,073
4	bbs.cqzg.cn	春秋中文社区	Discuz!	428,555
5	forums.crackberry.com	BlackBerry Forums	vBulletin V2	525,381
6	forums.gentoo.org	Gentoo Forums	phpBB V2	681,813
7	lkcn.net/bbs	英华论坛	IP.Board	180,692
8	techreport.com/forums	Tech Report	phpBB	65,083
9	www.redandwhitekop.com/forum	Liverpool FC Forum	SMF	138,963

We preferred nine forums (Table 2) among the 190 test forums for this assessment investigation. Eight of the nine forums are popular software packages used by many forum sites this is about 53 percent of forums powered by the 200 packages deliberate in this paper, and about 15 percent of all forums we have found.

Fig.- 2 Coverage comparison between the structure-driven crawler, iRobot, and FoCUS.



In this module, we report the results of the comparison between the structure-driven crawler, iRobot, and FoCUS. Although the structure-driven crawler is not a forum crawler, it could be utilized to forums. To make a more meaningful comparison, we used it to find page-flipping URL patterns in order to increase its coverage. As to iRobot, we re-implemented it. We permit the structure-driven crawler, iRobot, and FoCUS crawl each forum until no more pages could be retrieved. After that we counted how many threads and other pages were crawled, correspondingly.

ISSN: 2278 – 7798

V. Conclusion

In the paper we present tale method crawler which downloads and stores web pages, frequently for a web search engine. The rapid growth of internet poses more challenges to search for suitable link. We also symbolize the technique of FOCUS which are developed to extract only the relevant web pages of interested topic from the Internet. The design of FOCUS is capable to evaluate the text which found on a link with the input text file. The crawler uses pattern recognition and generates the number of times the input text exists in the text establish on a link. The information so generated gives an imminent in the efficiency of the pattern-matching. FoCUS constantly keeps on crawling the internet and finds any new internet pages that have been added to the web, pages that have been detached from the web. Due to growing and vibrant activity of the internet; it has become a confront to traverse the URLs in the web documents and to handle these URLs. We will take the seed URL as input and search with a keyword, the searching result is based on keyword and it will obtain the web pages where it will find that keyword.

Acknowledgment

We thanks to Software Department, Vee Eee Technology Solutions Pvt. Ltd, Which supported for implementation and testing for result in this research work.

REFERENCES

[1] S. Brin and L. Page, "The Anatomy of a Large-Scale Hypertextual Web Search Engine."Computer Networks and ISDN Systems,vol. 30, nos. 1-7, pp. 107-117, 1998.

[2] R. Cai, J.-M. Yang, W. Lai, Y. Wang, and L. Zhang, "iRobot: An Intelligent Crawler for Web Forums," Proc. 17th Int'l Conf. World Wide Web,pp. 447-456, 2008.

[3] Dasgupta, R. Kumar, and A. Sasturkar, "De-Duping URLs via Rewrite Rules," Proc. 14th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 186-194, 2008.

[4] Gao, L. Wang, C.-Y. Lin, and Y.-I. Song, "Finding Question-Answer Pairs from Online Forums," Proc. 31st Ann. Int'l ACMSIGIR Conf. Research and Development in Information Retrieval,pp. 467-474, 2008.

[5] N. Glance, M. Hurst, K. Nigam, M. Siegler, R. Stockton, and T.Tomokiyo, "Deriving Marketing Intelligence from Online Discussion," Proc. 11th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 419-428, 2005.

[6] Y. Guo, K. Li, K. Zhang, and G. Zhang, "Board Forum Crawling: A Web Crawling Method for Web Forum,"Proc. IEEE/WIC/ACM Int'l Conf. Web Intelligence, pp. 475-478, 2006.

[7] M. Henzinger, "Finding Near-Duplicate Web Pages: A Large-Scale Evaluation of Algorithms," Proc. 29th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 284-291, 2006.

[8] H.S. Koppula, K.P. Leela, A. Agarwal, K.P. Chitrapura, S. Garg, and A. Sasturkar, "Learning URL Patterns for Webpage De-Duplication," Proc. Third ACM Conf. Web Search and Data Mining, pp. 381-390, 2010.

[9] K. Li, X.Q. Cheng, Y. Guo, and K. Zhang, "Crawling Dynamic Web Pages in WWW Forums," Computer Eng., vol. 33, no. 6, pp. 80-82, 2007.

[10] G.S. Manku, A. Jain, and A.D. Sarma, "Detecting Near-Duplicates for Web Crawling," Proc. 16th Int'l Conf. World Wide Web, pp. 141-150, 2007.

[11] U. Schonfeld and N. Shivakumar, "Sitemaps: Above and Beyond the Crawl of Duty," Proc. 18th Int'l Conf. World Wide Web,pp. 991-1000, 2009.

[12] X.Y. Song, J. Liu, Y.B. Cao, and C.-Y. Lin, "Automatic Extraction of Web Data Records Containing User-Generated Content," Proc. 19th Int'l Conf. Information and Knowledge Management, pp. 39-48, 2010.

[13] V.N. Vapnik,The Nature of Statistical Learning Theory. Springer, 1995.

[14] M.L.A. Vidal, A.S. Silva, E.S. Moura, and J.M.B. Cavalcanti, "Structure-Driven Crawler Generation by Example," Proc. 29th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 292-299, 2006.

[15] Y. Wang, J.-M. Yang, W. Lai, R. Cai, L. Zhang, and W.-Y. Ma, "Exploring Traversal Strategy for Web Forum Crawling," Proc. 31st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 459-466, 2008.

[16] J.-M. Yang, R. Cai, Y. Wang, J. Zhu, L. Zhang, and W.-Y. Ma, "Incorporating Site-Level Knowledge to Extract Structured Data from Web Forums," Proc. 18th Int'l Conf. World Wide Web,pp. 181-190, 2009.

[17] Y. Guo, K. Li, K. Zhang, and G. Zhang, "Board Forum Crawling: A Web Crawling Method for Web Forum," Proc. IEEE/WIC/ACM Int'l Conf. Web Intelligence, pp. 475-478, 2006.

[18] M. Henzinger, "Finding Near-Duplicate Web Pages: A Large- Scale Evaluation of Algorithms," Proc. 29th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 284-291, 2006.

[19] H.S. Koppula, K.P. Leela, A. Agarwal, K.P. Chitrapura, S. Garg, and A. Sasturkar, "Learning URL Patterns for Webpage De- Duplication," Proc. Third ACM Conf. Web Search and Data Mining, pp. 381-390, 2010.

ISSN: 2278 – 7798

[20] K. Li, X.Q. Cheng, Y. Guo, and K. Zhang, "Crawling Dynamic Web Pages in WWW Forums," Computer Eng., vol. 33, no. 6, pp. 80-82, 2007.

[21] G.S. Manku, A. Jain, and A.D. Sarma, "Detecting Near-Duplicates for Web Crawling," Proc. 16th Int'l Conf. World Wide Web, pp. 141- 150, 2007.

[22] U. Schonfeld and N. Shivakumar, "Sitemaps: Above and Beyond the Crawl of Duty," Proc. 18th Int'l Conf. World Wide Web, pp. 991- 1000, 2009.

[23] X.Y. Song, J. Liu, Y.B. Cao, and C.-Y. Lin, "Automatic Extraction of Web Data Records Containing User-Generated Content," Proc. 19th Int'l Conf. Information and Knowledge Management, pp. 39-48, 2010.

[24] V.N. Vapnik, The Nature of Statistical Learning Theory. Springer, 1995.

[25] M.L.A. Vidal, A.S. Silva, E.S. Moura, and J.M.B. Cavalcanti, "Structure-Driven Crawler Generation by Example," Proc. 29th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 292-299, 2006.

[26] Y. Wang, J.-M. Yang, W. Lai, R. Cai, L. Zhang, and W.-Y. Ma, "Exploring Traversal Strategy for Web Forum Crawling," Proc. 31st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 459-466, 2008.

T.K.ARUNPRASATH, revised the BTech Information Technology from Anna University, Chennai, India, 2010. He is currently pursuing Master of Engineering in Computer Science and Engineering, A.S.L Pauls College of Engineering and Technology, Coimbatore, Tamilnadu, India. Research interests include fields of Networking and Data Mining.

DR. C. KUMAR CHARLIE PAUL, Principal, A.S.L Pauls College of Engineering and Technology, Coimbatore, Tamilnadu, India.

ISSN: 2278 – 7798