Volume 4, Issue 9, September 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

# Performance Enhancement of FoCUS Technique for Medical Information

**Namrata Bamrah***
Department of Computer Engg, DYPIET,
Pimpri, India

**Prof. B. S. Satpute**
Department of Computer Engg, DYPIET,
Pimpri, India

***Abstract— The web contains large amount of data with uncountable websites that is monitored by a tool or a program called Crawler. It is very difficult to retrieve the relevant information or data from the web. Forum threads show the information that is the target of Forum crawlers i.e it shows the contents of the posts. The layout or styles of forums is different from web and are powered by different forum software packages, to lead users from entry pages to thread pages i.e target pages forums have similar implicit navigation paths connected by specific URL types. Based on this observation, the web forum crawling problem is reduced to a URL-type recognition problem. In this paper, we present a method called FoCUS (Forum Crawler Under Supervision), supervised web-scale forum crawler. The main task of FoCUS is to crawl the relevant information from the forum based on predefined topic. FoCUS crawls the selected pages based on predefined set of topics or keywords. FoCUS uses multithreaded approach for crawling web forum pages which improves performance, and for recognizing URL type of forums FoCUS uses IT Regexes technique that is regular expression technique.***

***Keywords— Forum Crawling, FoCUS, Multithreading, IT Regex, URL Type***

## I. INTRODUCTION

Internet forum or web forum or message board is an online discussion site where users hold conversations in the form of messages which are posted on the forum known as discussions or post [1]. Web forums are used to request and exchange information with each other. As web contains millions of web pages, it is very difficult to obtain relevant information that the user has demanded from that particular search engine. For example Google crawls innumerable pages per day but it takes weeks to crawl the whole web because of unrelated and unwanted pages. So it is a difficult job to find relevant information, for this crawler is used.

The breadth first strategy crawler crawls the web and stores all the relevant data and show hyperlinks as a result, but because of this the database becomes too large to handle. If this drawback is handled then it's simple for the user to get his desired data and also the size of database can be reduced. So to avoid this drawback a crawler is needed that searches only the subset of the web not the whole web, but for this a crawler has to address two problems.

1. It should have good strategy for deciding which pages to download next.
2. It must have a reliable system that maintains and manages harmful results or effects of system crashes.

Due to detail of knowledge in forums, researchers are more interested in mining knowledge from them. Zhai and Liu [6], Yang et al. [10], and Song et al. [12] proposed a method to extract from forums the structured information. Glance et al. [3] tried from forum data to mine business intelligence. Zhang et al. [8] proposed algorithms in forums to extract expertise network. To retrieve data from forums, their content must be downloaded first. Generic crawlers [1] uses a breadth-first traversal strategy which is usually ineffective and inefficient for forum crawling, because of duplicate links and page-flipping links. Forums generally have several duplicate links pointing to a common page but with different URLs. A generic crawler follows these links and crawl several duplicate pages, making it inefficient. There is also a problem of entry URL discovery. Existing crawling methods such as Vidal et al. [4] and Cai et al. [8] are not effective because it doesn't have an entry URL.

The other challenges in forums are:

1. Detection of number of posts.
2. Classify topics.
3. Identify contents and users.

Our contributions of work in this paper are as follows:

1. We show that, FoCUS crawls the forum pages based on predefined topic and recognizes the url type.
2. We show that, how to learn regular expression patterns (IT regexes) that recognize the Index URLs and Thread URLs.

*© 2014, IJARCSSE All Rights Reserved* 

3. We show that, the learned patterns are effective and the resulting crawler is efficient.
4. It defines EIT path which allow more than one path.
5. We show that, FoCUS learns URL patterns across multiple sites.
6. Effective for large-scale forum crawling.
7. We show that, it increases Performance by using Multithreading approach.
8. We show that, it increases accuracy by content crawling rather than anchor text crawling based on predefined topic.

We show that, URL patterns would not be affected by a change in page structure.

## II. METHODS OF FORUM CRAWLING

### *A. Board Forum Crawling*

Yan Guo [3] proposed a new method of Board Forum Crawling to crawl Web forum. This method exploits the organized characteristics of the Web forum sites and simulates human behaviour of visiting Web Forums. This method starts crawling from the homepage, and then enters each board of the site, and then crawls all the posts of the site directly. Board Forum Crawling can crawl most meaningful information of a Web forum site efficiently and simply. But it crawls the website which has similar structure; because of this it is not suitable for large-scale crawling.

### *B. Structure-Driven Crawler Generation*

For learning regular expression patterns of URLs that lead a crawler from an entry page to target page, Structure-Driver Crawler Generation [4] technique is used. Target pages were found through comparing DOM trees of pages with a preselected sample target page. It is very effective but it only works for the specific site from which the sample page is drawn. If the Page structure changes the URL Pattern will be affected. In contrast, URL patterns would not be affected by a change in page structure in FoCUS System.

### *C. iRobot*

It is an intelligent crawler for Web Forums. The fundamental step in many web applications is web forum crawling problem, such as search engine and web data mining. Web forum crawling is not a trivial issue due to the in-depth link structure, the massive amount of duplicate pages and many invalid pages caused by login failure problems. For this, prototypes of an intelligent forum crawler is proposed and build known as iRobot [8], which has intelligence to grasp the content and therefore the structure of a forum site, and then decide the way to select traversal paths among different kinds of pages. The main drawback of this technique is its tree-like traversal path which does not allow more than one path and its URL location might become invalid when the page structure changes. iRobot does not deal with the frequent thread updating in forum. No clear segregation of page identification is carried out in iRobot and it cannot parse the crawled forum pages to separate replies in each post thread.

## III. PROPOSED SYSTEM

### *A. System Architecture*

FoCUS (FORUM CRAWLING UNDER SUPERVISION) is a supervised web-scale forum crawler. Fig. 1 below shows the overall architecture of FoCUS. It consists of two major parts: the learning part i.e a Training part and the online crawling part i.e a Testing part. In learning part (Training part), FoCUS first learns IT Regexes (Index – Thread Regular Expressions) from the URLs set of a particular forum. In online crawling part (Testing Part), FoCUS applies the learned IT regexes to crawl all threads/posts efficiently. FoCUS defines EIT (Entry – Index – Thread) path which specifies what type of links and pages the crawler should follow to reach thread pages. FoCUS defines URL patterns using IT Regex which would not be affected by a change in structure of the page. The user will first enter or select the topic whose posts the user wants to retrieve and then selects the forums from the list of forum links for crawling the posts. The pages will be downloaded first and then it will be parsed. User can select multiple forum links. Here multithreaded approach is implemented which increases the performance.

After crawling, it uses Index/Thread URL Detection technique to detect Index and Thread URLs of selected forums. On the detected Index/Thread URLs, IT Regexes is applied which will divide Index and Thread URLs separately. From the detected Thread Pages, the FoCUS detects related and unrelated Thread Pages based on predefined topic. Finally it will show the posts related to predefined keyword or topic and save it. The proposed system maintains a record of already crawled data. The proposed system maintains a record of already crawled data. In previous systems for retrieving the related posts based on predefined topic, the system finds the topic name in anchor text (title of the thread) rather in contents, which reduces the accuracy because it shows the limited related posts but increases the speed of crawling. Whereas FoCUS finds the topic name in the contents of the crawled URLs, which increase the accuracy because here depth crawling is done and more posts related to that particular topic is shown, but it reduces the speed of crawling.

This technique includes:

1. Index and Thread URLs Detection
2. Related and Unrelated Thread URLs
3. IT Regex
4. Post Detection

*© 2014, IJARCSSE All Rights Reserved* 

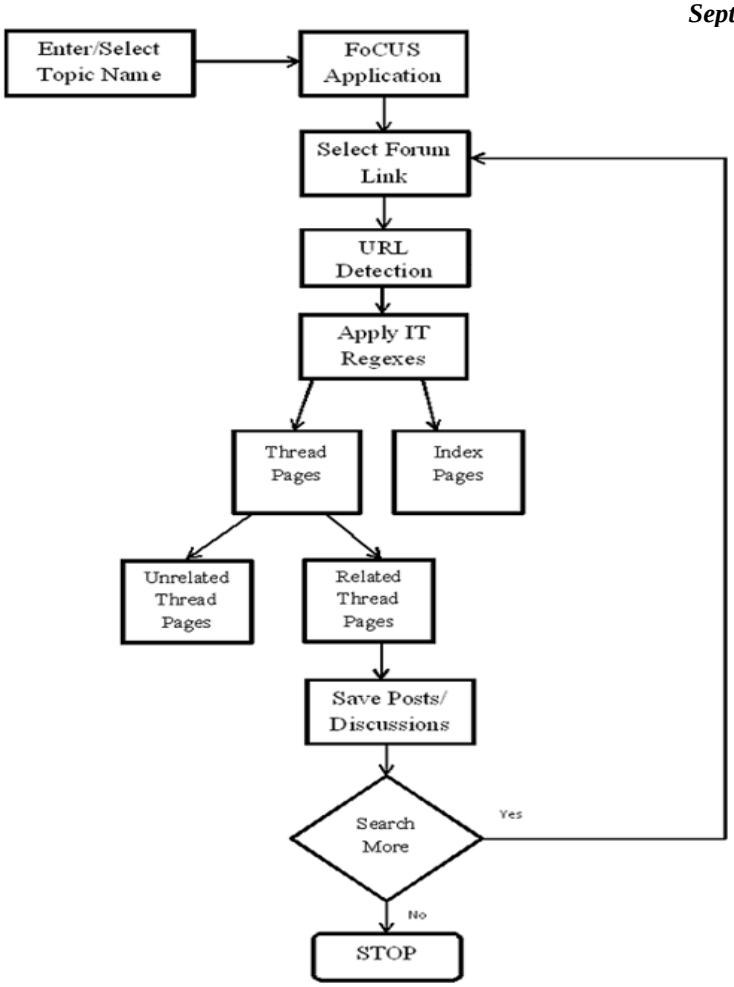


Fig. 1 FoCUS System Architecture

*Index and Thread URLs Detection*

An Index URL is a URL that is on an entry page or index page. The destination page of an index URL is another index page (list of threads). A thread URL is a URL that is on an index page. The destination page of the thread URL is a thread page (posts). The difference between index URLs and thread URLs is the type of their destination pages. Therefore, to decide the page type of a destination page some method is needed. The index pages and thread pages each have their own typical layouts. The following algorithm detects Index and thread URLs:

**Algorithm: DetectIndexAndThreadURL**

Input: A group of URLs

Output: Index and Thread URLs

1. Links = a group of URLs;
2. foreach (theURI in Links)
3. theURI = dequeue(Links)
4. if(theURI matches the Thread Regex)
5. lstThreadURLs = Add the URL to Thread URLs list
6. else
7. if (theURI matches the Index Regex)
8. lstIndexURLs = Add the URL to Index URLs List
9. end if
10. end if
11. end

*IT Regex*

Vidal et al. [4] applied URL string generalization method, the proposed scheme do not use their method because it is too strict and difficult to understand. FoCUS uses IT Regexes for URL detections. An Index-Thread (IT) regex is regular expressions that are used to recognize index URLs and thread URLs. IT regex is what FoCUS aims to learn in Training part and applies directly in online crawling in testing part. The learned IT regexes are site specific, and there are two IT regexes in a site: one for recognizing Index URLs and the other is for recognizing Thread URLs. Instead of URL locations FoCUS learns URL patterns to discover new URLs. Thus, it does not need to classify new pages in crawling and would not be affected by a change in page structures.

*© 2014, IJARCSSE All Rights Reserved* 

Each IT Regex contains 2 elements:

1. URL Type
2. URL Pattern (REGEX)

Consider the following URLs in Table 1 from the forum site, http://www.drugs-forum.com. The first and second URL are the Index URLs. Third and Fourth URL are the Thread URLs.

TABLE I: SAMPLE URLS

| Sr. No. | URLs |
|---|---|
| 1. | http://www.drugs-forum.com/forum/forumdisplay.php/f=43 |
| 2. | http://www.drugs-forum.com/forum/forumdisplay.php/f=424 |
| 3. | http://www.drugs-forum.com/forum/showthread.php/?t=31648 |
| 4. | http://www.drugs-forum.com/forum/showthread.php/?t=31651 |

TABLE III: IT REGEX FOR URLS

| URL TYPE | URLs |
|---|---|
| Index | http://www.drugs-forum.com/\w+/\w+.php/f=\d+ |
| Index | http://www.drugs-forum.com/\w+/\w+.php/f=\d+ |
| Thread | http://www.drugs-forum.com/\w+/\w+.php/\?t=\d+ |
| Thread | http://www.drugs-forum.com/\w+/\w+.php/\?t=\d+ |

where, \d+ represents the page number that is digits;
\w+ represents the string for the Index/Thread Page.

For the URLs in Table I the regular Expression (REGEX) are formed as in Table II above.

*Related and Unrelated Thread URLs*

The Related and Unrelated Thread URLs are decided from all crawled Thread URLs based on predefined Topic entered by the user. First the predefined topic is searched in the group of Thread URL contents. If the URL content contains the topic, that URL is added to the related URL list or else it will be added in unrelated URL list. FoCUS increases accuracy by content crawling rather than anchor text crawling.

**Algorithm: RelatedThreadList**

Input: lstThreadURLs: a group of Thread URLs
Output: lstThreadURLsMatched: a group of Related Thread URLs

1. lstThreadURLs = all Thread URLs
2. foreach (url in lstThreadURLs)
3. url=dequeue(lstThreadURLs)
4. URLContents = extracts corresponding contents
5. if (URLContents contains Topic)
6. lstThreadURLsMatched = add URL to Related list;
7. lblThreadRelCount = show the count of URLs;
8. else
9. lstThreadURLs.Items = keep the URLs in List;
10. lblThreadNRelCount = show the count of URLs;
11. end if
12. end

### *B. Performance Analysis*

**PRECISION** is a fraction of crawled (retrieved) pages that are relevant to topic. Precision can be defined by the following equation:

$$\textbf{Precision} = Pr / Tn$$

where, Pr = number of pages visited that are relevant and
Tn = total number of pages visited.

**RECALL** is a fraction of relevant pages that have been retrieved. Recall can be defined by the following equation:

$$\textbf{target\_recall} = \frac{|Ptg \cap Pcr|}{|Ptg|}$$

Where, Ptg = target pages and Pcr = crawled pages

**EFFECTIVENESS** measures the percentage of the thread pages among all pages crawled of a forum. It is defined as below:

*© 2014, IJARCSSE All Rights Reserved* 

$$\text{Effectiveness} = \frac{Tp}{(Tp + Op) \times 100\%}$$

where, Tp = all threads pages; Op = other pages

## IV. RESULTS & DISCUSSIONS

The FoCUS crawls the pages of forums based on entered or selected topic by the user. Performance of the system is checked on the basis of three metrics - precision, recall and Effectiveness. The data set for the FoCUS system is the list of forums and the list of topics. The below Fig.2 shows the result screen of FoCUS system. The first column displays the list of topics, the second column displays the list of different forums and the third column displays the crawled posts or discussions of the particular crawled topic.

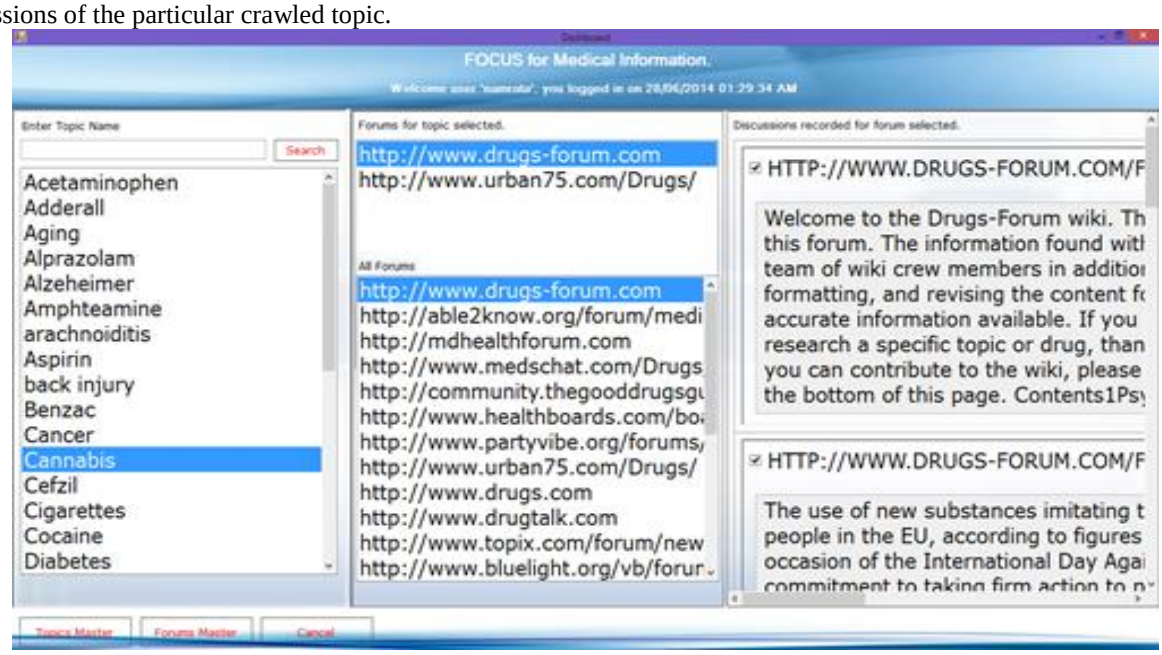


Fig. 2 Results Screen of FoCUS System

## V. CONCLUSIONS

FoCUS is a supervised forum crawler. It crawls the forums, recognizes type of URLs and shows how to leverage implicit navigation paths of forums, i.e., EIT path, and designed methods to learn IT regexes. FoCUS uses multithreading approach which improves performance of the system. FoCUS increases accuracy by crawling the contents of the forums for a particular topic rather than crawling the anchor text of the forum which gives more relevant data. FoCUS can effectively learn EIT path from few annotated forums. FoCUS can effectively collect index URL and thread URL training sets and learn IT regexes from the training sets. These learned regexes are applied directly in online crawling.

Results on few forums show that FoCUS can apply the learned knowledge to a large set of forums and still achieve a very good performance. Though the proposed method is targeted at forum crawling, the implicit EIT-like path also applies to other sites, such as community QuesAns sites and blog sites.

## REFERENCES

[1] J. Jingtian Jiang, Xinying Song, Nenghai Yu, ,"FoCUS: Learning to Crawl Web Forums," IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 25, NO. 6, JUNE 2013

[2] Chakrabarti, S., M. van den Berg, and B. Dom (1999, May). "Focused crawling: A new approach to topic-specic Web resource discovery," In WWW 99: Proceeding of the 8th International Conference on World Wide Web, New York, NY, pp. 16231640. Elsevier.

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

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

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

[6] Y. Zhai and B. Liu, "Structured Data Extraction from the Web based on Partial Tree Alignment," IEEE Trans. Knowledge Data Eng., vol. 18, no. 12, pp. 1614-1628, Dec. 2006.

[7] 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

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

*© 2014, IJARCSSE All Rights Reserved* 

[9] 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. Intl ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 459-466, 2008.

[10] 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 Intl Conf. World Wide Web, pp. 181- 190, 2009.

[11] 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.

[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 Intl Conf. Information and Knowledge Management, pp. 39-48, 2010.

*© 2014, IJARCSSE All Rights Reserved* 