ISSN 23499842(Online), Volume 1, Special Issue 1(NCEE'15), May 2015
International Journal of Innovative Trends and Emerging Technologies

A KNOWLEDGE BASED SYSTEM FOR LEARNING FOR CRAWL WEB FORUMS USING FOCUS

Dr. S. ANITHAA. PhD, A.REKHA
Professor (Dept of Cse), PG Scholar Dept of CSE,
Sri Ramanujar Engineering College, Vandalur, Chennai-127.
rekhasvcoe@gmail.com

Abstract- The World-Wide-Web (WWW) is growing exponentially and has become increasingly difficult to retrieve relevant information on the web. The rapid rising of the WWW poses unprecedented scaling challenges for general purpose crawlers and search engines. In this paper, we present Forum Crawler Under Supervision (FoCUS), a supervised web-scale forum crawler. The goal of FoCUS is to crawl relevant forum content from the web with minimal overhead, this crawler is to selectively seek out pages that are relevant to a predefined set of topics, rather than collecting and indexing all accessible web documents to be able to answer all possible ad-hoc queries. FoCUS continuously keeps on crawling the web and finds any new web pages that have been added to the web, pages that have been removed from the web. Due to growing and dynamic nature of the web; it has become a challenge to traverse 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 web pages where it will find that keyword.

1.1 OVERVIEW OF THE PROJECT

A web crawler is a relatively simple automated program, or script that methodically scans or "crawls" through Internet pages to create an index of the data it's looking for; these programs are usually made to be used only once, but they can be programmed for long-term usage as well. There are several uses for the program, perhaps the most popular being search engines using it to provide webs surfers with relevant websites. In addition to the above two challenges, there is also a problem of entry URL discovery. The entry URL of a forum points to its homepage, which is the lowest common ancestor page of all its threads. Our experiment "Evaluation of Starting from Non-Entry URLs" in a crawler starting from an entry URL can achieve a much higher performance than starting from non entry URLs. assumed that an entry URL is give Other users include linguists and market researchers, or anyone trying to search information from the Internet in an organized manner. Alternative names for a web crawler include web spider, web robot, bot, crawler, and automatic indexer. Crawler programs can be purchased on the Internet, or from many companies that sell computer software, and the programs can be downloaded to most computers. To harvest knowledge from forums, their content must be downloaded first. However, forum crawling is not a trivial problem. Generic crawlers, which adopt a breadth-first traversal strategy, are usually ineffective and inefficient for forum crawling. This is mainly due to two non crawler friendly characteristics of forums 1) duplicate links and uninformative pages and 2) page-flipping links. A forum typically has many duplicate links that point to a common page but with different URLs , e.g., shortcut links pointing to the latest posts or URLs for user experience functions such as "view by date" or "view by title." A generic crawler that blindly follows these links will crawl many duplicate pages, making it inefficient. A forum also has many uninformative pages such as login control to protect user privacy or forum software specific FAQs. Following these links, a crawler will crawl many uninformative pages. Though there are standard-based methods such as specifying the "rel" attribute with the "no follow", Robots Exclusion Standard (robots.txt), and Sitemap for forum operators to instruct web crawlers on how to crawl a site effectively, we found that over a set of nine test forums more than 47 percent of the pages crawled by a breadth-first crawler following these protocols were duplicates or uninformative. Web crawlers may operate one time only, say for a particular one-time project. If its purpose is for something long-term, as is the case with search engines, web crawlers may be programmed to comb through the Internet periodically to determine whether there has been any significant changes. If a site is experiencing heavy traffic or technical difficulties, the spider may be programmed to note that and revisit the site again, hopefully after the technical issues have subsided A Web crawler starts with a list of URLs to visit, called the *seeds*. As the crawler visits these URLs, it identifies all the hyperlinks in the page and adds them to the list of URLs to visit, called the *crawl frontier*. URLs from the frontier are recursively visited according to a set of policies. The large volume implies that the crawler can only download a limited number of the Web pages within a given time, so it needs to prioritize its downloads. The high rate of change implies that the pages might have already been updated or even deleted.

The number of possible crawlable URLs being generated by server-side software has also made it difficult for web crawlers to avoid retrieving duplicate content. Endless combinations of HTTP GET (URL-based) parameters exist, of

which only a small selection will actually return unique content. For example, a simple online photo gallery may offer three options to users, as specified through HTTP GET parameters in the URL. If there exist four ways to sort images, three choices of thumbnail size, two file formats, and an option to disable user-provided content, then the same set of content can be accessed with 48 different URLs, all of which may be linked on the site. This mathematical combination creates a problem for crawlers, as they must sort through endless combinations of relatively minor scripted changes in order to retrieve unique content.

1.2 PROBLEM DEFINITION

To retrieve relevance forum data effectively and efficiently, we should first understand the characteristics of most forum URL. In general, content of a forum is stored in a database. When a Web forum service receives a user request, it dynamically generates a response page based on some pre-defined templates with page request time, has value and URL address. The almost forum site is connected as a very complex graph with many links among various pages. Due to these reasons, it's very difficult to extract exact URL which has relevance information about user query. First, we have to indentify duplicate pages (or content) with different Uniform Resource URLs will be generated by the service for different requests such as "view by date" or "view by title." Second, a long post divided into multiple pages usually results in a very deep navigation. Sometimes a user has to do tens of navigations if he/she wants to read the whole thread, and so does a crawler. Finally, it display exact URL which has relevance information of user query.

1.3 OBJECTIVES OF THE PROJECT

II. Literature Survey:

BOARD FORUM CRAWLING: A WEB CRAWLING METHOD FOR WEB FORUM

- This method exploits the organized characteristics of the Web forum sites and simulates human behavior of visiting Web Forums.
- The 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.
- This method used in a real project, and 12000 Web forum sites have been crawled successfully.

INTERNET forums (also called web forums) are important services where users can request and exchange information with others. The need of the hour is to make the process of searching the internet for information more and more efficiently. The goal of FoCUS is to crawl relevant forum content from the web. With the size of the internet increasing exponentially, the volume of data to be crawled also proportionally increases, as a result of which it becomes increasingly necessary to have appropriate crawling mechanisms in order to make crawls efficient. . A FOCUS is a computer program that browses the Internet in a methodical automated manner. The FOCUS crawler typically crawls through links grabbing content from websites and adding it to search engines indexes. We are concentrating on FOCUS crawler which search for the relevant web pages based on the keyword we give. Actually it forms a hierarchy of links. The crawler on the particular web page for a particular keyword, which we give as, input. It will search for the link on that seed URL and after that switch to that link and find another link on that web page but it should match with the keyword, it will do like that until it reach the limit that we set.

1.4 FEATURES

- its first validate the user query form forum database,
- Its only extract standard URL page who have proper URL format .
- Our method only extract exact URL (relevance information) based on user query.
- Here, its creates ranking of index pages based user query information however our techniques does not consider famous or popularity web pages.
- Finally it display query related URL pages to user.

FINDING QUESTION-ANSWER PAIRS FROM ONLINE FORUMS

Online forums contain a huge amount of valuable user generated content. In this paper we address the problem of extracting question-answer pairs from forums. Question-answer pairs extracted from forums can be used to help Question Answering services (e.g. Yahoo! Answers) among other applications. We propose a sequential patterns based classification method to detect questions in a forum thread, and a graph based propagation method to detect answers for questions in the same thread. Experimental results show that our techniques are very promising.

III.Existing Framework:

EXISTING SYSTEM

All the major search engines have highly optimized crawling system, although working and

details of documentation of this system are usually with their owner. It is easy to build a crawler that would work slowly and download few pages per second for a short period of time. In contrast, it's a big challenge to build the same system design, I/O, network efficiency, robustness and manageability. Every search engine is divided into different modules among those modules crawler module is the module on which search engine relies the most because it helps to provide the best possible results to the search engine.

PROPOSED SYSTEM

We are concentrating on focus crawler which search for the relevant web pages based on the keyword we give. Actually it forms a hierarchy of links. The crawler on the particular web page for a particular keyword, which we give as, input. It will search for the link on that seed URL and after that switch to that link and find another link on that web page but it should match with the keyword, it will do like that until it reach the limit that we set.

IV.Proposed Framework:
SYSTEM DESIGN ANALYSIS

4.1 ARCHITECTURAL DIAGRAM

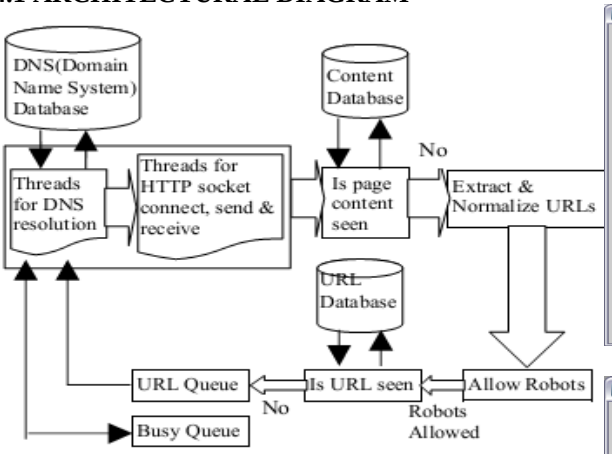
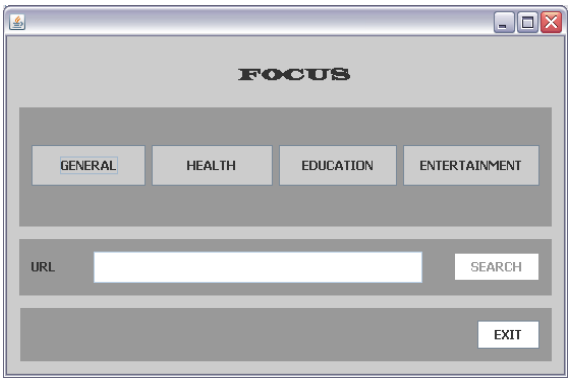
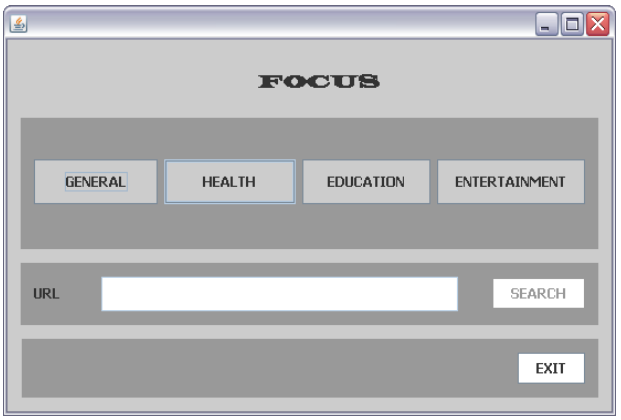
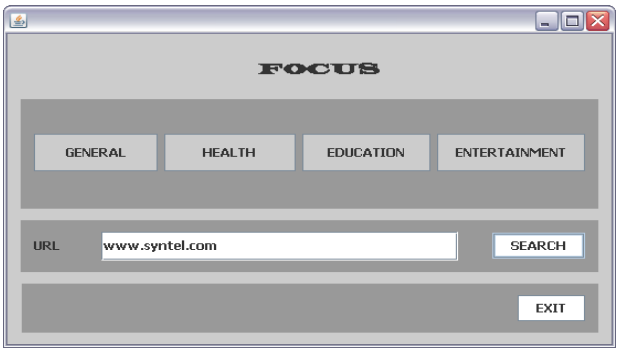


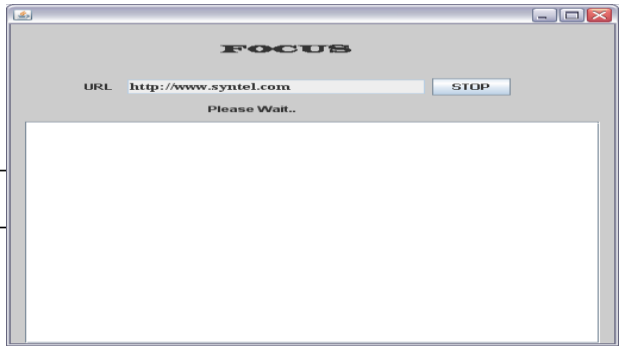
Fig 4.1 Architecture diagram

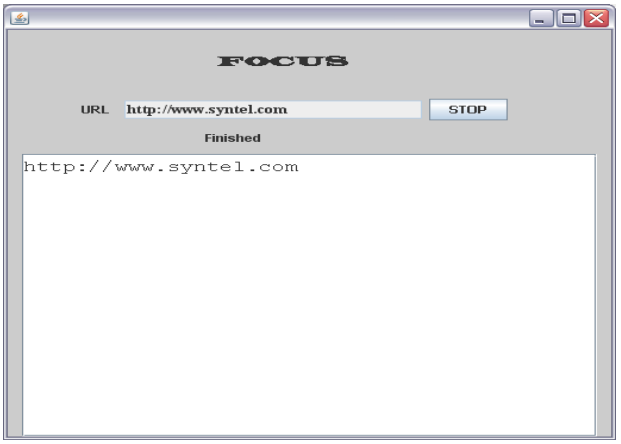
4.2 DESIGN OF SCREENSHOTS

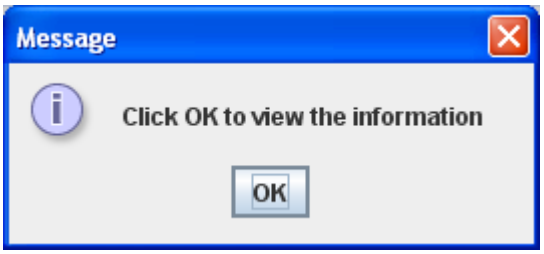












V Performance Evaluation:

MODULE DESCRIPTION

1. Read URL:

We are concentrating on focus ontology which search for the relevant web pages based on the keyword we give. Actually it forms a hierarchy of links. The web information on the particular web page for a particular keyword, which we give as, input. It will search for the link on that seed URL and after that switch to that link and find another link on that web page but it should match with the keyword, it will do like that until it reach the limit that we set. But it may be possible that it will not found the number of links that we set before. Then it shows that the web page is not having any further link for that particular keyword. While fetching the links the user profiles also make sure that it should fetch only the unique links, means that it should not revisit the same link again and again. Finally , when we finished with the links, we will give one txt file as input and run the three pattern matching algorithm.

2. Pattern Recognition:

Here with pattern we mean only text. Pattern matching is used for syntax analysis. When we compare pattern matching with regular expressions then we will find that patterns are more powerful, but slower in matching. A pattern is a character string. All keywords can be written in both the upper and lower cases. A pattern expression consists of atoms bound by unary and binary operators. Spaces and tabs can be used to separate keywords. Text mining is an important step of knowledge discovery process. It is used to extract hidden information from not-structured or semi-structured data. This aspect is fundamental because much of the web information is semi-structured due to the nested structure of HTML code, much of the web information is linked, and much of the web information is redundant. Web text mining helps whole knowledge mining process of mining, extraction and integration of useful data, information and knowledge from the web page content. Pattern recognition is applied on the web information like this , When we start the retrieval it will give me the links related to the keyword. It will then read the web pages that are extracted from the links and while it will read the web page it will extract only the content. Here content means only the text that is available on the web page. It should not include images, tags, and buttons. The extracted content should be stored in some file. But it should not include any HTML tags.

3. Identification Process:

This process will identify the required url is whether right kind of link or wrong kind link. It will identify the url, protocol link also for retrieve the relevant web page for user requesting. It's used to omit bad urls while user requesting web pages. Bad urls are identified by pattern of protocol occur on the relevant web pages on the server side.

4. Downloading Process:

After completion of all process the downloading will started. It will start to downloading requesting url link of users need. After three checking process only it will downloaded the relevant link for users request. It will working efficiently to users, the requested link will retrieve all web pages through the ontology model.

VI.CONCLUSION

A crawler is a program that downloads and stores web pages, often for a web search engine. The rapid growth of World Wide Web poses challenges to search for the most appropriate link. FOCUS is developed to extract only the relevant web pages of interested topic from the Internet. The designed FOCUS is capable of comparing the text 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 found on a link. The information so generated gives an insight in the efficiency of the pattern-matching. FoCUS continuously keeps on crawling the web and finds any new web pages that have been added to the web, pages that have been removed from the web. Due to growing and dynamic nature of the web; it has become a challenge to traverse 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 web pages where it will find that keyword.

6.1 FUTURE ENHANCEMENT

The FOCUS performs the pattern recognition using three algorithms independently and generates the number of performed by each algorithm. The information so generated gives an insight in the efficiency of the pattern-matching algorithm. The FOCUS designed is using only one technique of text mining i.e. pattern recognition. The FOCUS can further be extended to use other text mining techniques. There by making a crawler more intelligent and better equipped in finding copyright infringement.

VIII References:

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]. A. 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]. C. 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.