ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

Automation of URL Discovery and Flattering Mechanism in Live Forum Threads

T.Nagajothi[1], M.S.Thanabal[2]

PG Student, Department of CSE, P.S.N.A College of Engineering and Technology, Tamilnadu, India [1]

Associate Professor, Department of CSE, P.S.N.A College of Engineering and Technology, Tamilnadu, India [2]

ABSTRACT: The Internet is an effective media for information sharing and propaganda broadcasting. The extraordinary growth of the Internet has resulted in due attention on web crawling techniques in recent years. In spite of huge leaps in communication, storage and work out power in recent years, hidden URL identifiers always fight back to keep up with web content generation and modification. Also there is no specific pattern matches to identify the proper URLs.Hence there is a need to focus on attempting to speed up the traversal process, to increase the production of high quality pages and to allocate appropriate tribute to different content along a traversal path. We intend to associate an automation that is aware of traversing the contents dynamically. This automated traversal enhances the performance of the system substantially. The proposed system involves Differential content extraction technique in monitoring the Forums and manipulates the possibility of migrating the forum threads to the next level or relevant groups.

KEYWORDS: Web crawler, Forum crawling, Information retrieval, page classification, URL type, Key word search.

I. INTRODUCTION

Web Forum (also known as Internet Forum) is a web application for making discussion and posting user query. Web forum may refer to either the particular community or a specific forum dealing with a distinct topic. It is used to know about the user's opinion and to understand what their expectations. To reap, data from Forum their content should be downloaded first. A Web crawling is a process which can collect forum data automatically and store it in a database. The data can be used for big data analysis and web content mining. In the existing system there is no clear path segregation is done.

It uses tree like traversal strategy, which takes only one path from entry page to thread page. It doesn't maintain record for already crawled forum site since it is a time consuming process. The main disadvantage of existing system is

- It takes only one path from entry page to thread page.
- Entry URL is discovered using human inspection.
- It doesn't make use of differential content extraction.

A crawler tried to create an automation engine which crawl the dynamic content. Cleanup of data and moving to the related link are done by the crawler. In the proposed system uses differential content extraction instead of entire system scanning which will enhance the performance of the system. This is done by the number of link options + page indexes. Scanning the entire pages through KMP algorithm. Proposed system uses a record of already crawled data which is used for future analysis.

The major contributions of this paper are as follows

- We create an automation engine to crawl dynamic content.
- Capturing the related link, moving the content to the appropriate site is the scope of the project.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

- It uses Forum package for crawling the related link.

II.RELATED WORK

Vidal et al proposed a method for learning regular expression pattern of URLs which lead a crawler from entry page to target page. Thread pages where found through presampled pages. It is used only for specific forum site since it is not applied for large forum crawling. Proposed system learns URL patterns from multiple sites so it can be used for large forum crawling

Wang et al make use of a method to address the traversal selection problem. It uses the skeleton link and page flipping link to identify the related link and to traverse the content to appropriate page. Skeleton links are used to crawl only the valuable pages not the Un informative pages. Page flipping links are used to download the thread from different forum sites. The demerit is that it doesn't deal with optimizing the records. Proposed technique uses page flipping pattern matching to deal with un informative pages and to obtain the robustness.

Vidal et al uses aims to learn a forum crawler with minimum human intervention by sampling pages, bunch live and finding a traversal path by a spanning tree rule. However, the traversal path choice procedure needs human scrutiny. It uses the method of tree like traversal it doesn't allow more than one path from entry page to thread page. It uses URL location information to identify the new URL, if the page structure changes then the URL location information become invalid. Proposed crawler uses URL patterns to discover the new URL it doesn't deal with the newly crawled pages and if the page structure changes then there is no change in the URL.

Guo et al uses the method to identify and traverse the URL.They used some heuristic rules to identify the URL.Since the rules are specific and only can be applied for particular software package. So it uses different rules for different packages. Proposed technique uses top down key word search algorithm to find the exact path for the user query. Then KMP (Knuth–Morris–Pratt) search algorithm is used to find the appropriate thread link and traverse the content to appropriate thread page.

Entry URL discovery is the trivial problem in existing crawler, since they assume that the URL at the home page is the entry URL and it may be varies for different forum packages and sites. Without entry URL the crawler is less effective. Another important work is detection of duplicate links. Existing techniques like content based duplicate detection and URL based duplicate detection are used. Content based duplicate detection is less effective since it prone to less bandwidth.URL based duplicate detection is not efficient for URL with same text. By using the URL patterns in proposed system duplicate pages can be removed. It is efficient and more robust.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

III. PROPOSED WORK

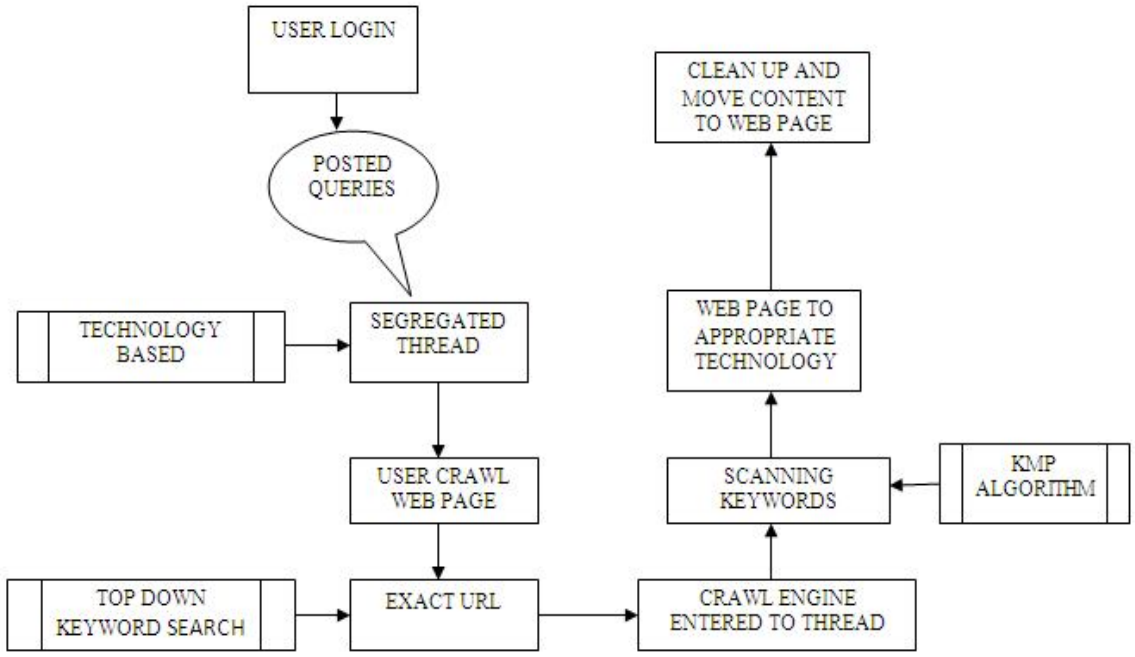


FIG 1: A low level architecture, of crawler

FIG 1 shows the low level architecture of the crawler. The user come with query and posts it to the forum page. The crawler makes use of the forum package and fined the appropriate entry URL using the entry URL discovery phase. Entry URL is discovered using the heuristic rules. Based on this baseline it uses the keyword such as forum, board, community, bbs and discus. If any one of the keyword is found then the URL is said to be entry URL.After finding the entry URL crawler try to find the index and thread URL from the entry page.

These pages are finding through the index/thread URL phase. Index URL is a URL which is on the Index page or the thread page and its destination is the index page and its text is the board title. A thread URL is a URL which is on the Index page or Thread its destination page is the thread page. Based on the characteristics of the page type and the time stamp ordering index and thread URLs are identified. Based on the page classifier index and thread pages are classified it uses SVM classifier to identify the pages using some heuristic rules.

After finding the particular URL Exact path of the individual thread is identified using top down keyword search algorithm. Once it has identified the individual thread KMP search algorithm is used to scan the entire pages and to find the correct page link which matches the keyword. The keyword which is fined will be compared with the existing dataset. If the key word matches then the page is moved to appropriate site. Then the unrelated links has been removed and the space is allocated to the new pages. Finally the result is given to the appropriate forum site.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

PROPOSED ALGORITHM:

TOP DOWN KEY WORD SEARCH ALGORIHM:

- Keyword search algorithm is an algorithm for finding an item with specified properties among a collection of items.
- The items may be stored individually as records in a database.
- A Keyword search looks for words anywhere in the record.
- The key words are searched top to down in the database.

This algorithm used to find the appropriate URL.

KMP SEARCH ALGORITHM:

The Knuth–Morris–Pratt string searching algorithm (or KMP algorithm) searches for occurrences of a "word" W within a main "text string" S by employing the observation that when a mismatch occurs, the word itself embodies sufficient information to determine where the next match could begin, thus bypassing re-examination of previously matched characters.

```
algorithm kmp_search:
  input:
    an array of characters, S (the text to be searched)
    an array of characters, W (the word sought)
  output:
    an integer (the zero-based position in S at which W is found)
define variables:
    an integer, m ← 0 (the beginning of the current match in S)
    an integer, i ← 0 (the position of the current character in W)
    an array of integers, T (the table, computed elsewhere)

  while m + i < length(S) do
    if W[i] = S[m + i] then
      if i = length(W) - 1 then
        return m
      let i ← i + 1
    else
      let m ← m + i - T[i]
      if T[i] > -1 then
                    let i ← T[i]
      else
        let i ← 0
   (if we reach here, we have searched all of S unsuccessfully)
  return the length of S
```

EXAMPLE:

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

- Knuth–morris–pratt string searching algorithm (or kmp algorithm) searches for occurrences of a "word" w within a main "text string" s by employing the observation that when a mismatch occurs
- The word itself embodies sufficient information to determine where the next match could begin, thus bypassing re-examination of previously matched characters.
- We proceed by comparing successive characters of w to "parallel" characters of s, moving from one to the next if they match. However, in the fourth step, we get s[3] is a space and w[3] = 'd', a mismatch.
- Rather than beginning to search again at s[1], we note that no 'a' occurs between positions 0 and 3 in s except at 0; hence, having checked all those characters previously.
- we know there is no chance of finding the beginning of a match if we check them again. Therefore we move on to the next character, setting m = 4 and i = 0.

IV. EXPERIMENTAL RESULTS

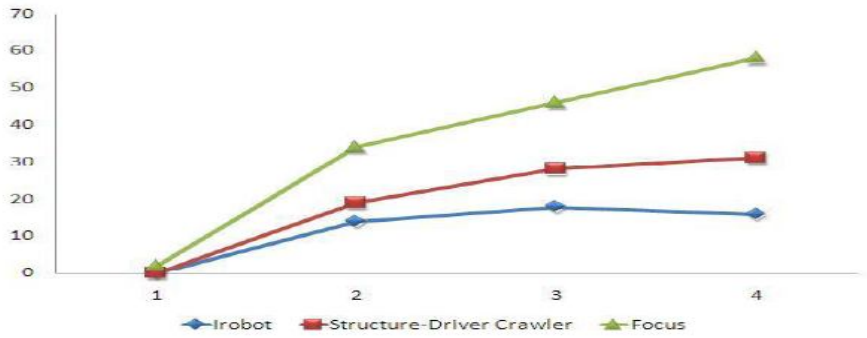
We have implemented our proposed techniques. We compared the efficiency of the proposed crawler with existing crawler such as iRobot, structure driven crawler and generic crawler. The obtained results were satisfying in the advanced crawling with respect of relvency, effectiveness, coverage and time of crawling the forum package.

4.1 Entry URL identification:

Existing crawler assumes that entry URL is given. Entry URL discovery is compared with heuristic baseline. We consider the 20 forum package to evaluate the efficiency of Entry URL discovery. Compared to the existing techniques the percent of precision and recall achieved is 99 percent. The crawler will be used for longer operation.

4.2 Efficiency comparison:

We evaluated the efficiency of the crawler using the number of pages crawled and the time spent for crawling. The results are evaluated under the metric of coverage over five forum package. We limited the method to sample at most k pages ,where k varies from 100 to 5000.then we let the methods crawling forum packages using learned knowledge.fig shows the efficiency comparison with existing crawlers.



To estimate the time spent on crawling, we ran these systems on a machine with two dual core 2.20 GHz CPU and 64 GB memory and 64 bit windows 8 os .The maximum time the crawler sent was 1,100 secs and the minimum time the crawler sent was 125 secs.This relatively less compared to existing crawler.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

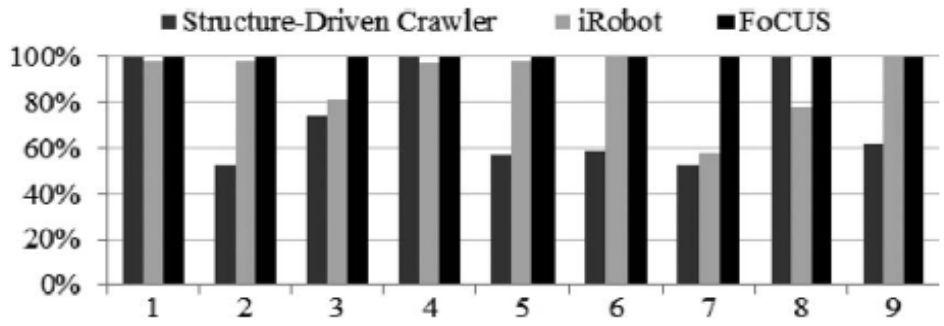
Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

4.3 Relevancy of web pages:

Relevancy of the retrieved web pages is greater than the existing crawler since it uses KMP search algorithm to find the appropriate web page. Relevancy is compared by the retrieval of the URL. In the existing crawler relevancy is about 60% and our crawler achieved more than 90% in relevant URL retrieval.



4.4 Online crawling

We report the results of the comparison between the existing crawlers To make a more meaningful comparison, we adapted it to find page-flipping URL patterns in order to increase its coverage. We let the all the crawler will crawl each forum packages until no more pages could be retrieved. After that we counted how many threads and other pages were crawled, respectively .As showed earlier, starting from non entry URLs will yield lower coverage. In online crawling, we let crawlers start from non entry URLs and entry URLs, respectively .The results of existing crawler starting from non entry URLs are much worse than starting from entry URLs. The coverage is lower than 30 percent in average in existing one but our crawler achieved more 90 percent of coverage by using entry URL.

V. CONCLUSION AND FUTURE WORK

A supervised forum crawler has been proposed and implemented. It will automatically crawl the dynamic content. In which the Entry URL is discovered automatically and the efficiency is compared using 200 forum sites. SVM page classifier is used to identify the index and thread URL that gives more accuracy in classification of pages. Differntial content extraction is a technique which uses two algorithm top down key word search algorithm and KMP search algorithm. These algorithms are used to find the appropriate site and to remove the irrelevant forum site from the record. In future the crawler can be used for all type of web pages also can be applied in cloud analysis and big data analysis.

ACKNOWLEDGEMENT

It is my pleasure to acknowledge Mr.J.Venkatesan Prabhu, Head, Kaashiv- InfoTech, Chennai for his support in implementation part and for his guidance throughout this course of work.

REFERENCES

[1] forummatrix.://www.forummatrix.org/index.php
[2] hotscripts.http://www.hotscripts.com/index.php
[3]internetforum. http://en.wikipedia.org/wiki/internet_forum
[4] 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.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

[5] Y. Guo, K. Li, K. Zhang, and G. Zhang. Board Forum Crawling: a Web Crawling Method for Web Forum. *Proc. 2006 IEEE/WIC/ACM Int'l Conf. Web Intelligence*, pp. 475-478, 2006.
[6] C. Gao, L. Wang, C.-Y. Lin, and Y.-I. Song. Finding Question-Answer from Online Forums. *Proc. 31st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval*, pp. 467-474, 2008.
[7] Wang.Y, Yang.J.-M, Lai.W, Cai.R, Zhang.L, and Ma.W.-Y. ,'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.
[8] 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.
[9] 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.
[10]Jingtian Jiang, Xinying Song, Nenghai Yu, and Chin-Yew Lin, FoCUS: Learning to Crawl Web Forums,*IEEE transactions on knowledge and data engineering, vol. 25, no. 6, June 2013.*
[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] K. Li, X.Q. Cheng, Y. Guo, and K. Zhang. Crawling Dynamic Web Pages in WWW Forums. *Computer Engineering*, vol. 33, no. 6, pp. 80- 82,2007.
[13] 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.
[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.