ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization, Volume 3, Special Issue 1, February 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHNICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

Multi Keyword Web Crawling Using Ontology in Web Forums

P.Jananipriya[1], Dr.P.Vivekanandan[2], Mrs.A.Anitha[3]

II-M.E CSE, Department of CSE, Park College of Engineering and Technology, Coimbatore, Tamilnadu, India [1]

HOD/CSE, Department of CSE, Park College of Engineering and Technology, Coimbatore, Tamilnadu, India[2]

Assistant Professor, Department of CSE, Muthayammal College of Engineering, Rasipuram, Tamilnadu, India[3]

ABSTRACT--Internet forums are online discussion sites where people can do conversations in the form of messages. Each forum is having sub-forums and it contains different topics based on the people's discussion. Crawling is the initial and the most important step during the Web searching procedure. Existing system presents a supervised web-scale forum crawler called Forum Crawler under Supervision (FoCUS). The goal of the FoCUS is to collect the forum pages with minimum overhead. During Crawling, the existing system uses only the single keyword method to crawl the web pages. It does not discover new threads and also does not refresh the crawled threads in a timely manner. The above two problems are rectified in the proposed system by using Ontology concept for Multi Keyword web crawling and Temporal database for discovering new threads. To improve the efficiency the proposed new crawler collects web pages for indexing from the web. By using Ontology concept, the crawling efficiency will be increased and also page coverage will be increased.

Index Words: Crawler, Forum, Ontology, Thread page

I. INTRODUCTION

A crawler is a system for bulk downloading of pages. They are the main component of the search engines. Every crawler will download the pages and index them for user request queries and find the match for the user queries. The basic web crawling algorithm is simple: Given a set of seed Uniform Resource Locators (URLs), a crawler downloads all the web pageand mines, all the hyperlinks contained in the pages and continuously downloads the web pages mentioned by these hyperlinks. Despite the seeming simplicity of this basic algorithm, web crawling has many inherent challenges: scale, content selection tradeoffs, social obligation and adversaries.

II. RELATED WORK

In generic crawlers [2], breadth first search and depth first search algorithms are used traversal path in those pioneer papers. But, both are inefficient because they won't reject the duplicate and uninformative pages. To overcome this above problem a new intelligent crawler has been proposed which can skip the duplicate and uninformative pages. For this a method called spanning tree algorithm is used for traversal path. Then, repetitive region based algorithm is used to discover the sitemap and the presampled pages are grouped into multiple clusters. This uninformativeness is automatically estimated and optimal traversal started not only by its URL patterns but also with the locations of the links in the pages. It collects the pages in optimal navigation and maintains the maximum coverage of pages. It keeps around 95% page relations in crawling. It is still not enough for object level storage. IRobot skips the invalid and uninformative pages. In future, important work is to design the repository for forum archiving.

While collecting the web forum pages [3], elimination of duplicate pages is an important step for all the crawlers. But, the elimination not yet received attention in the generic crawler. For this above problem, one of the methods has been proposed to detect the near duplicate during the forum crawling. The technique used for this above mentioned problem is Sim hash technique is used to solve near-duplicates. This shows well

ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization, Volume 3, Special Issue 1, February 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHNICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

detection of near duplication pages for 64-bit fingerprint web pages. It made the advantages of storage size decreases, network bandwidth has not been wasted because this detecting near duplication. Detection process has done only for small sized fingerprints. It can't handle the large size of finger prints. Its future work may be categorize the web pages and detect the web pages.

Due to rapid advance in technology and web proliferation, creating a web search engine today is very different from years ago. In that some of challenges involved when using additional information present in hypertext to produce better search results. To build a practical large-scale system this can exploit the additional information present in hypertext. And another problem of how to effectively deal with the uncontrolled hypertext collections where anyone can publish anything they want.

In web search [4], presence of duplicate documents in the web will affect the indexing and relevance. The above two are the important thing to web search. To overcome the duplication problem, learning URL patterns have been discovered. It used the technique called URL preprocessing, Pairwise generation and Rule generalization algorithm. It achieves 2 times more reduction ratio compared to previous work. It also achieves 69% of reduction ratio for 100% of rules. Its advantages deals with that it reduces duplicate pages. In future, to generalize the source and target pages in iterative fashion.

In web crawling, another problem is that the duplication of URLs. When the user starts to crawls the URLs there may be duplication of URLs. This duplication deals with wastage of time and memory. Hence A. Dasgupta, R. Kumar, and A. Sasturkar [5] proposed rewriting URLs as rules. In this, Inductive learning framework is used to identify different URL with similar text. Finally, this technique shows that 60% of reduction of duplication of URLs. This process improves the efficiency of the processing of crawling. It is a simple framework but proven its capacity in large scale. In future, this method can be used to capture wider set of rules.

Forum crawling is the challenging task due to some of major issues. Some issues such as in-site link structures and login controls. And generic crawlers will download all the pages which consists uninformative pages and invalid pages. Yid Wang et. Al [6] proposed exploring traversal strategy for web crawling. This proposed system to explore automatically explores the traversal path which directs the crawling of given target forum. It uses two techniques such as sitemap reconstruction and traversal strategy which consists of two steps such as identification of skeleton links and detection of page flipping links. It shows that this proposed method skeleton identification, crawls almost 1.53 times valuable pages than a generic crawler and 1.32 times valuable pages than the structure driven. Regarding to crawling time also it shows its valuable performance. Based on the page flipping links, it shows its results as more accurate than the previous works. In future, to optimize the crawling schedule to incrementally update the archived forum content and how to parse the crawled forum pages to separate replies in each post thread.

III. SYSTEM ANALYSIS

1. Existing work

Vidal et al. [13] proposed a method to learn regular expressions of URLs that move a crawler from entry page to target page. With a preselected sample target page the DOM trees of pages could be compared to find the target page.. It is suitable for specific site. It is not suitable for large scale crawling. Cai et al. [2] proposed concept called iRobot. It showed the method to learn the crawler with minimum human intervention. The Human intervention is needed for the selection of traversal path. Follow up work by Wang et al. [14] proposed an algorithm for the traversal path selection problem. By showing skeleton links and page-flipping links, this system can achieve the page coverage.

The above all the existing systems deals with that the forum pages download from the web forindexingpurpose in search engine. The Proposed system learns the pattern automatically based on the training set**.** After that, search engine can easily extract the pages from the database for the user given request. When crawler collecting the web pages from the web,

ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization, Volume 3, Special Issue 1, February 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHNICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

crawler uses single keyword to collect the web pages. This method will not cover all the web pages with the same semantics of the given key word. Then, another one drawback is new threads are not updated in the search engine database.

2. Proposed System

Considering existing systems, all the systems can collect the web pages for indexing in the search engines. Instead of using single keyword matching method, proposed system uses multi keyword matching method called Ontology. Another problem in existing system is, it unable to discover the dynamic threads, so this can be done by using temporal database in proposed system.

It collects the pages based on the relevant content pages. It covers most of the pages based on the semantic of given semantic. Its accuracy up to 98% pages collection. Proposed system learns and classifies new pages in crawling and would effective change in page structures.

IV. SYSTEM OVERVIEW

1. Entry URL discovery

This module is used to detect the Entry URL of any given forum page.Entry URL discovery algorithm is used to discover entry URL. It also uses heuristic rule to find the URL.After detection, Entry URL can be used for online crawling.

Algorithm EntryUrlDiscovery

Input: url: a URL pointing to a page from a forum
Output: entry_url:Entry URL of this forum
Step 1: b_url-GetNaiveEntryUrl(url); //baseline
Step 2: p=Download(url);
Step 3: urls=Extract outgoing URLs in p that start with b_url;
Step 4: samp_urls=Randomly sample a few URLs from urls;
Step 5: Add the host of url into samp_urls; //observation(2)
Step 6: foreach u in samp_urls do
Step 7: p=Download(u);
Step 8: urls=urls ∩ {outgoing URLs in p starting with b_url};
Step 9: end foreach
Step 10: let entry_url be b_url, index_urls be φ, count be 0;
Step 11: foreach u in urls do
Step 12: if u is in index_urls continue; //observation
Step 13: p=Download(u);
Step 14: i_urls=Detec index URLs in p;
Step 15: Index_urls=index_urls U i_urls;
Step 16: if count < |i_urls|
Step 17: count=|i_urls|;
Step 18: entry_url=u;
Step 19: end if
Step 20: end foreach
Step 21: return entry_url;

2. Index and Thread URL Detection

This module is used to detect Index and Thread URLs.Index and Thread URL detection algorithm is used.Support vector machine page type classification is used to classify the pages. Index and thread URL detection algorithm is used to detect the URLs. Index page have many links and every links have another index or a thread page. If there is an index page further page collection is proceed or else if that link is a thread page crawling of that link will be stopped. After the URLs are detected, URLs can be saved in training set.

Algorithm IndexURLANDThreadURLDetection

Input: sp: an entry page or index page
Output: it_group: a group of I ndex/thread URLs
Step 1: let it_group be φ ;data
Step 2: url_groups=collect URL groups by aligning HTML DOM tree of sp;
Step 3: foreach ug in url_gruops do
Step 4: ug.anchor_len=Total anchor text length in ug;
Step 5: end foreach
Step 6: it_group=arg_max(ug.anchor_len) in url_groups;

ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization, Volume 3, Special Issue 1, February 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHNICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

Step 7: it_group.DstPageType=Majority page type of the destination pages og URLs in ug;
Step 8: if it_group.DstPageType is INDEX_PAGE
Step 9: it_group.urlType=INDEX_URL;
Step 10: else if it_group.DstPageType is THREAD_PAGE
Step 11: it_group.urlType=THREAD_URL;
Step 12: else
Step 13: it_group= φ;
Step 14: end if
Step 15: return it_group;

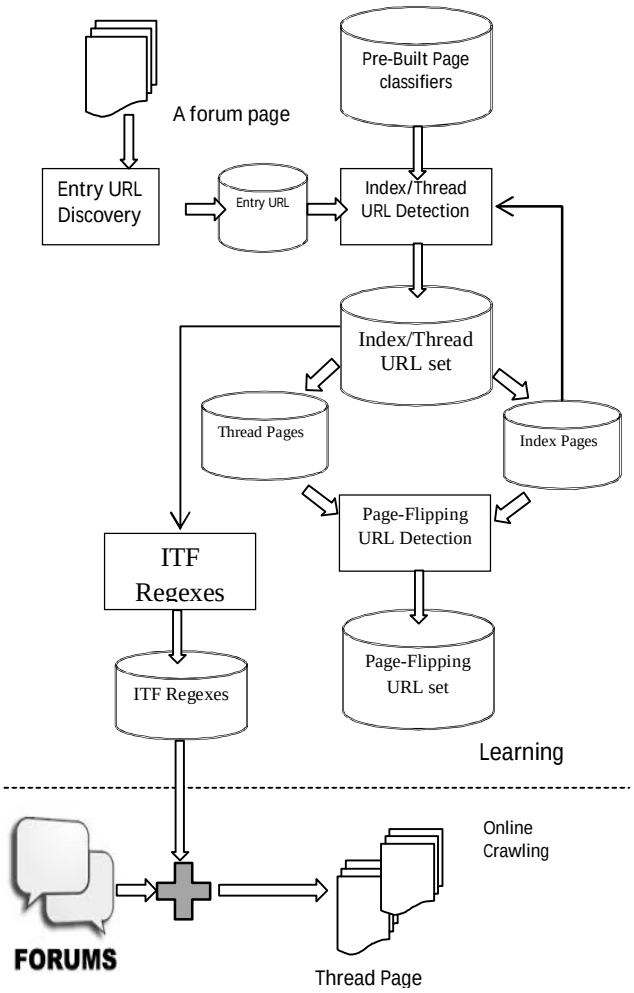

Fig. 1. System architecture

3. Page Flipping URL Discovery

This module detects Page flipping URLs from index and thread pages.This module uses Page Flipping URL detection algorithm.URLs can be saved in training set.

4. Learning ITF Regexes

In this module, all the collected training set URL must be learnt to find the specific URL pattern. Based on this specific pattern Online crawling will be done. For this pattern learning, Vidal et al [13] string generalization is not used. Because proposed system training sets are constructed automatically so, that method is very difficult to learn. So, proposed system followed method which is followed by Koppula et al [8]. It will overcome negative examples also. In this method, patterns are refined again and again. When the pattern reaches final refinement the it will be used for the online crawling.

Algorithm PageFlippingUrlDetection

Input: sp: an index page or thread page
Output: pf_group: a group of page_flipping URLs
Step 1: let pf_group be φ;
Step 2: url_groups=Collect URL groups by aligning HTML DOM tree of sp;
Step 3: foreach ug in url_groups do
Step 4: if the anchor texts of ug are digit strings
Step 5: pages=Download(URLs in ug);
Step 6: if pages have the similar layout to sp and ug appears at same location of pages as in sp
Step 7: pf_group=ug;
Step 8: break;
Step 9: end if
Step 10: end if
Step 11: end foreach
Step 12: if pf_group is φ
Step 13: foreach url in outgoing URLs in sp
Step 14: p=Download(url);
Step 15: pf_url=Extract URL in p at the same location as url in sp;

ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization, Volume 3, Special Issue 1, February 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHNICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

Step 16: if pf_url exists and pf_url.anchor==url.anchor and pf_url.UrlString != url.UrlString
Step 17: Add url and cand_url into pf_group;
Step 18: break;
Step 19: end if;
Step 20: end foreach
Step 21: end if
Step 22: pf_group.UrlType=PAGE_FLIPPING_URL;
Step 23: return pf_group;

Based on the pages collected in training sets, it will be classified using SVM classifier. This proposed will not need any strong classifier. It classifies the pages and gives accuracy up to 93 percentages.

5. Multi keyword web crawling

Crawling will be done based on the Ontology concept. Here keyword can be taken as multi keyword for the crawling procedure. Each given keyword have multiple synonyms, based on that, pages will be collected to the database. Collected pages can be displayed as index to improve the search timing of the search engine and increase the user friendly. It improves the page coverage.

Another important one is updating of timely occurring threads in a forum page. Existing system didn't concentrate on that. But the proposed system will overcome those drawbacks by updating the timely occurring threads in a forum page. So, the database will be always updated.

V. CONCLUSION AND FUTURE WORK

FoCUS, a supervised forum crawler. The proposed system reduced the forum crawling problem to a URL type recognition problem and the system showed the navigation path of forums such as EIT and plannedmethods to learn ITF regexes clearly.

FoCUS can efficiently learn knowledge of EIT path from as few as five forums. The system also shows that FoCUS can successfully apply learned forum crawling knowledge on seven forums to automatically collect all the URL training sets such as Index, thread, and Page Flipping URLs and learn ITF regexes from that collected training sets. These learned regular expressions can be applied directly in online crawling. First, the system must be given training and apply the trained sets to online crawling procedure. After this the procedure can be applied to other forum sites and the results applicable to numerous forum sites. All the existing system expects an entry URL but, FoCUS can start from any page of a forum. Proposed system briefly evaluated our relevance computation algorithm which is based over onto crawler concept. The proposed done this in a small and controlled environment so that the functioning of the algorithm could be clearly visualized and experienced. Test results on seven forums show that FoCUS is indeed very effective and efficient and it outperforms in accuracy compared to the intelligent forum crawler called iRobot. After finishing that, accuracy of FoCUS deals with 93% of pages.

In future, in this web crawling based mechanism, the system support multiple keywords use of Ontology. Hence, by using Ontology concept the accuracy of crawling must be increased. Most of the pages from the web can be collected based on the user relevant information. Based on the semantic meaning of the given keyword, the forum pages get collected. It is possible to measure the performance of a search by understand user interest and information relevant. Also, it is possible to try to handle the JavaScript generated URLs. Then, it can discover new threads and refresh crawled threads in a dynamic manner. The initial results of applying a FoCUS-like crawler to other social media are very promising. Above crawler can be used to conduct more comprehensive experiments to further verify this approach and improve upon it.

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,"

ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization, Volume 3, Special Issue 1, February 2014

International Conference on Engineering Technology and Science-(ICETS'14)

On 10th & 11th February Organized by

Department of CIVIL, CSE, ECE, EEE, MECHNICAL Engg. and S&H of Muthayammal College of Engineering, Rasipuram, Tamilnadu, India

Proceedings 17th International Conference World Wide Web, pp. 447-456, 2008.

[3] A. Dasgupta, R. Kumar, and A. Sasturkar, "De-Duping URLs via Rewrite Rules," Proceedings 14th ACM SIGKDD International Conference 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," Proceedings 31st Ann. International ACM SIGIR Conference 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," Proceedings 11th ACM SIGKDD International Conference 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," Proceedings IEEE/WIC/ACM International Conference Web Intelligence, pp. 475-478, 2006.

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

[8] JingtianJingtian Jiang, Xinying Song, Nenghai Yu, "FoCUS:LearningtoCrawlWebForums" IEEE Transactions on knowledge and Data Engineering, vol.25 no.6 pp.1293-1306.

[9] H.S. Koppula, K.P. Leela, A. Agarwal, K.P. Chitrapura, S. Garg, and A. Sasturkar, "Learning URL Patterns for Webpage DeDuplication," Proceedings Third ACM Conference Web Search and Data Mining, pp. 381-390, 2010.

[10] 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.

[11] G.S. Manku, A. Jain, and A.D. Sarma, "Detecting Near-Duplicates for Web Crawling," Proceedings 16th International Conference World Wide Web, pp. 141150, 2007.

[12] U. Schonfeld and N. Shivakumar, "Sitemaps: Above and Beyond the Crawl of Duty," Proceedings 18th International Conference World Wide Web, pp. 9911000, 2009.

[13] X.Y. Song, J. Liu, Y.B. Cao, and C.-Y. Lin, "Automatic Extraction of Web Data Records Containing User-Generated Content," Proceedings 19th International Conference Information and Knowledge Management, pp. 39-48, 2010.

[14] M.L.A. Vidal, A.S. Silva, E.S. Moura, and J.M.B. Cavalcanti, "Structure-Driven Crawler Generation by Example," Proceedings 29th Ann. International ACM SIGIR Conference 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," Proceedings 31st Ann. International ACM SIGIR Conference 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," Proceedings 18th International Conference World Wide Web, pp. 181190, 2009.

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

[18] J. Zhang, M.S. Ackerman, and L. Adamic, "Expertise Networks in Online Communities: Structure and Algorithms," Proceedings 16th International Conference World Wide Web, pp. 221-230, 2007.

[19] L. Zhang, B. Liu, S.H. Lim, and E. O'Brien-Strain, "Extracting and Ranking Product Features in Opinion Documents," Proceedings 23rd International Conference Computational Linguistics, pp. 1462-1470, 2010.