IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

Effective Crawling In Web Forum

D. Dhivya [1], R. Venkadeshan [2], D. Vidhya [3]

PG Scholar, Computer Science and Engineering, Chettinad College of Engineering and Technology, Karur, India[1]

Professor, Computer Science and Engineering, Chettinad College of Engineering and Technology, Karur, India[2]

PG Scholar, Computer Science and Engineering, K.S.R College of Engineering, Namakkal, India[3]

Abstract: The Main Objective of EC web is to crawl relevant forum content from the web with minimal overhead. Forum threads contain information content that is the target of forum crawlers. Each forum have different layouts or styles and are powered by different forum software packages, they always have similar implicit navigation paths connected by specific URL types to lead users from entry pages to thread pages. We reduce the web forum crawling problem to a URL-type recognition problem. Training sets are created by learning accurate and effective regular expression patterns. We have applied this knowledge on unseen URL's and identified the type of that URL. After the classification all crawled URL's are stored in a log. URL log is used to identify strong and weak URL's by eliminating the duplicate URL's from the URL log. Effectiveness of the strong URL will be measured finally.

Keywords: Effective Crawling, Web Forum, URL Type Recognition Module, Crawling Module.

I. INTRODUCTION

Forum is the place where the people can shares their knowledge with others. According to Forum Matrix, Forum is the right place to clarify the technical doubts and Forum Matrix is here to help you decide which forum is the best for your needs. According to Wikipedia, An Internet forum or message board is an online discussion site where people can hold conversations in the form of posted messages.

Internet forums (also called web forums) are important services where users can request and exchange information with others. For example, the Trip Advisor Travel Board is a place where people can ask and share travel tips. Due to the richness of information in forums, researchers are increasingly interested in mining knowledge from them. We extracted structured data from forums. We identified question and answer pairs in forum threads. We proposed methods to extract and rank product features for opinion mining from forum posts. We tried to mine business intelligence from forum data. The authors proposed algorithms to extract expertise network in forums.

The work of the web crawler in our venture is to recover whatever number applicable pages as would be prudent from the web. The important pages will be recovered for the URL that is given as the information. All the substance of the discussion pages will be recovered. The recovered substance will be again URLs just.

Those URLs will be of different classes. The different classifications incorporates list URL, page URL, solid URL, frail URL and so forth. In our venture, we defeat the URL sort distinguish issue by distinguishing the kind of the URL that is recovered by the crawler. . By eliminating the weak URLs we can get the URLs which is having the relevant information for the user. All of this is accomplished by utilizing the regex. Finally the assessment of the URL will be performed.

II. RELATED WORKS

In the existing system, the forums have different layouts or styles and are powered by different forum software packages; they always have similar implicit navigation paths connected by specific URL types to lead users from entry pages to thread pages. There is a huge web forum crawling problem.

Existing framework which proposed a system for taking in general statement examples of URLs that lead a crawler from a one page to target pages. Target pages were found through contrasting DOM trees of pages and a preselected specimen target page. It is extremely powerful yet it works for the particular site from which the example page is drawn. The same procedure must be rehashed each time for another site.

A later and more far reaching deal with gathering slithering is irobot intends to immediately take in a discussion crawler with least human intercession by examining pages, grouping them, selecting enlightening bunches by means of a useful measure, and discovering a traversal way by a spreading over tree calculation. Nonetheless, the traversal way choice method obliges human examination. Proposed a calculation to address the traversal way determination issue. They presented the thought of skeleton connection and page-flipping connection. Skeleton connections are "the most essential connections supporting the structure of a gathering site." Importance is controlled by useful and scope measurements. Page-flipping connections are resolved utilizing connectivity metric. By distinguishing and just emulating skeleton connections and page-flipping connections, they indicated that irobot can accomplish adequacy and scope. Consistent with our assessment, its testing method and useful estimation is not hearty and its tree-like traversal way does not permit more than one way from a beginning page hub to a same close page hub.

Another related work is near-duplicate detection. Forum crawling also needs to remove duplicates. But content

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

based duplicate detection is not bandwidth efficient, because it can only be carried out when pages have been downloaded. URL-based duplicate detection is not helpful. It tries to mine rules of different URLs with similar text. However, such methods still need to analyze logs from sites or results of a previous crawl. In forums, index URLs, thread URLs, and page-flipping URLs have specific URL patterns. Thus in this paper, by learning patterns of index URLs, thread URLs, and page flipping URLs and adopting a simple URL string de duplication technique, ECWEB can avoid duplicates without duplicate detection. Drawbacks of the Existing System The learned patterns are ineffective and inefficient.

III. PROPOSED SYSTEM

In the proposed system, we crawl the relevant forum content from the web with minimal overhead. Forum threads contain information content that is the target of forum crawlers. In the proposed system, we evaluate the effectiveness and efficiency of the Web crawler. The web crawler crawls the similar forum information from the web. This project also overcomes the URL Type recognition problem. The effectiveness of the web crawler means that given a number of retrieved pages, how many of them are valuable and informative. The efficiency of the web crawler means how fast a crawler can retrieve a given number of valuable pages. By eliminating the weak URLs we can get the URLs which is having the relevant information for the user.

Advantages of the Proposed System
- Learned Patterns are efficient
- Non Duplicate links
- Forum crawling is reduced to a URL type recognition problem.

IV. OVERVIEW OF THE SYSTEM

The Overview of the web crawler in our project is to retrieve as many relevant pages as possible from the internet. The relevant pages will be retrieved for the URL that is given as the input. All the contents of the forum pages will be retrieved. The retrieved contents will be again URLs only. Those URLs will be of various categories. The various categories includes index URL, page URL, strong URL, weak URL etc. In our project, we overcome the URL type recognition problem by identifying the type of the URL that is retrieved by the crawler. All of this is achieved by using the regex, pattern learning and so on. Finally the evaluation of the URL will be performed. After the evaluation we can filter the weak URLs and get strong URLs which have the relevant information.

Links between an entry page and an index page or between two index pages are referred as index URL. That an index URL is a URL that is on an entry or index page its destination page is another index page; its anchor text is the board title of its destination page. Thread URL, Links between an index page and a thread page are referred as thread URLs.

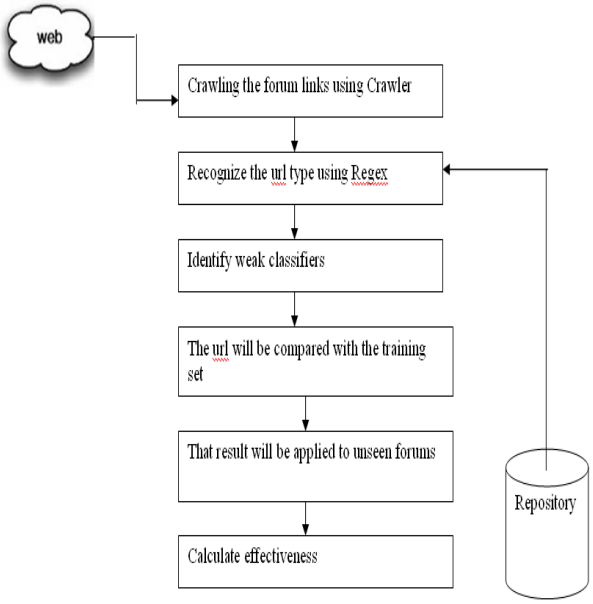

Figure 1: Architecture of proposed methodology

Page Flipping Links connecting multiple pages of a board and multiple pages of a thread are referred as page flipping URLs. All of this is achieved by using the regex, pattern learning and so on. Finally the evaluation of the URL will be performed. After the evaluation we can filter the weak URLs and get strong URLs which have the relevant information.

V. MODULES DESCRIPTION

i. CRAWLING MODULE
ii. URL TYPE RECOGNITION MODULE
iii. IDENTIFY WEAK CLASSIFIERS
iv. COMPARISON MODULE
v. EFFECTIVENESS MODULE

i) CRAWLING MODULE
In this module, Web crawling is a well-studied crisis with still ongoing challenges. A survey of the field of Web archiving and archival Web crawling is available. A goal-directed, crawler crawls the Web URL's according to a predefined set of web site topics, and thus influences the crawler performance not based on the configuration of Web applications as is our intend, but on the content of Web pages. The main functionality is crawling the web forum links using Crawler. The crawler retrieves all the associated information provided by the user. Besides duplicate links & uninformative pages, a long forum board or thread is usually divided into multiple pages which are linked by page-flipping links. When a dynamic page is requested, the web server first looks at the page's source code and if any server-side scripting code exists, it will process them and generate static HTML result. When processing of the full page has been completed, web server sends only pure HTML code to the web visitor's browser. Generic crawlers process each page individually and ignore the relationships between such pages. These relationships should be preserved while crawling to facilitate downstream tasks such as page wrapping and

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

content indexing. A page URL can contain more than one "?" character. When this happens, search engine spiders will have difficult time to index the resulted page. If the page has only one "?" character, major search engine spiders can crawl that page well.

ii) URL TYPE RECOGNITION MODULE

In this module, we identify the type of the url. Links between an entry page and an index page or between two index pages are referred as index URLs. Links between an index page and a thread page are referred as thread URLs. Links connecting multiple pages of a board and multiple pages of a thread are referred as page flipping URLs. Web forum in the form of a directed graph consisting of vertices (Web pages) and directed arcs (links between different Web pages). Furthermore a path analysis is performed to provide an optimal traversal path which leads the extraction process in order to avoid duplicate and invalid pages. The default behavior of the web crawler is to be as polite as possible. This page was crawled by Google. The length of its query parameter is 4 characters. There are many other examples on the internet that have more characters and were crawled successfully. The maximum number of characters that can be accepted by Google is unknown. At least now we can say Googlebot is able to crawl dynamic pages that have one query parameter and the number of characters in the parameter can be 4. This is enforced by requiring that only one crawler thread should be accessing an individual web server at any one time. This prevents multiple crawler threads from overloading a web server. It is implemented by mapping individual servers to specific crawler threads.

iii) IDENTIFY WEAK CLASSIFIERS MODULE

In this module we are going to identify the weak classifiers are the urls that does not have the relevant information for the crawled set of results. If you have a requirement to keep your index of a particular web site (or sites) as up-to-date as possible, you could create a specific collection for this area. You could then create a separate collection for the rest of your content which may not change as often or where the update requirements are not as stringent. This larger collection could be updated over a longer time period. By using a Meta collection you can then combine these collections so that users can search all available information. By identifying and only following skeleton links and page-flipping links, they showed that Analyzing weak URL can achieve effectiveness and coverage. ECWEB learns page type classifiers directly from a set of annotated pages based on this characteristic. According to our evaluation, its sampling strategy and in formativeness Estimation is not robust and its tree-like traversal path does not allow more than one path from a starting page node to a same ending page node.

iv) COMPARISON MODULE

In this module, we compare our entry URL discovery method with a heuristic baseline. We then, compare our system with other existing methods in terms of effectiveness and coverage. ECWEB is that index URL, thread URL, and page-flipping URL can be detected based on their layout characteristics and destination pages; and forum pages can be classified by their layouts. This knowledge about URLs and pages and forum structures can be learned from a few annotated forums and then applied to unseen forums. The WebCrawler will check the secondary store specified in this parameter and not download content from the web which hasn't changed. When a web collection is created the Funnel back administration interface will insert the correct location for this parameter, and it will not normally need to be edited manually. Increasing the number of crawlers (threads) will increase throughput, as will decreasing the delay between requests. The latter is specified in milliseconds, with a default delay of one quarter of a second. We do not recommend decreasing this below 100ms.

v) EFFECTIVENESS MODULE

In this Final module we are going to find out the effectiveness and coverage measure. Effectiveness measures the percentage of thread pages among all page crawled of a forum; coverage measures the percentage of crawled thread pages to all retrievable thread pages of the forum. We would like to have 100% effectiveness and 100% coverage when all retrievable threads of a forum are crawled and only thread pages are crawled. A crawler can have high effectiveness but low coverage and low effectiveness and high coverage. The generic crawler started from the entry URL and a randomly selected non-entry URL respectively. It stopped when no more pages could be retrieved. We repeated this experiment with different non-entry URLs.

The effectiveness of the URL is calculated by the formula,

$$\text{Effectiveness} = \frac{\#\text{Crawelled threads}}{\#\text{threads in all}} \times 100\%$$

4/7/20164/7/2016 ALGORITHMS

Index URL And Thread URL Detection Algorithm

Input: sp: an entry page or index page
Output: it_group: a group of index/thread URLs

1. let it_group be ϕ;data
2: URL_groups = Collect URL groups by aligning HTML DOM tree of sp;
3: foreach ug in URL_groups do
4: ug.anchor_len = Total anchor text length in ug;
5: end foreach
6: it_group = arg max(ug.anchor_len) in URL_groups;
7: it_group.DstPageType = Majority page type of the destination pages of URLs in ug;
8: if it_group.DstPageType is INDEX_PAGE
9: it_group.URLType = INDEX_URL;
10: else if it_group.DstPageType is THREAD_PAGE
11: it_group.URLType = THREAD_URL;
12: else
13: it_group = ϕ;
14: end if
15: return it_group;

Page Flipping URL Detection Algorithm
Input: sp: an index page or thread page

DOI 10.17148/IJARCCE.2016.53239

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

Output: pf_group: a group of page-flipping URLs

1: let pf_group be ϕ;
2: URL_groups = Collect URL groups by aligning HTML DOM tree of sp;
3: foreach ug in URL_groups do
4: if the anchor texts of ug are digit strings
5: pages = Download(URLs in ug);
6: if pages have the similar layout to sp and ug appears at same location of pages as in sp
7: pf_group = ug; 8: break;
9: end if
10: end if
11: end foreach
12: if pf_group is ϕ
13: foreach URL in outging URLs in sp
14: p = Download(URL);
15: pf_URL = Extract URL in p at the same location as URL in sp;
16: if pf_URL exists and pf_URL.anchor == URL.anchor and pf_URL.URLString != URL.URLString
17: Add URL and cand_URL into pf_group;
18: break;
19: end if
20: end foreach
21: end if
22: pf_group.URLType = PAGE_FLIPPING_URL;
23: return pf_group;

Entry URL Discovery Algorithm

Input: URL: a URL pointing to a page from a forum

Output: entry_URL:Entry URL of this forum

1: b_URL = GetNaiveEntryURL(URL); //baseline
2: p = Download(URL);
3: URLs =Extract outgoing URLs in p that start with b_URL;
4: samp_URLs = Randomly sample a few URLs from URLs;
5: Add the host of URL into samp_URLs; //observation (2)
6: foreach u in samp_URLs do
7: p = Download(u);
8: URLs = URLs □ {outgoing URLs in p starting with b_URL }; //observation (1)
9: end foreach
10: let entry_URL be b_URL, index_URLs be ϕ,count be 0;
11: foreach u in URLs do
12: if u is in index_URLs continue; //observation (3)
13: p = Download(u);
14: i_URLs = Detect index URLs in p;
15: index_URLs = index_URLs □ i_URLs;
16: if count < |i_URLs| //observation (4)
17: count = |i_URLs|;
18: entry_URL = u;
19: end if
20: end foreach
21: return entry_URL;

VI. CONCLUSION

Finally this project concludes by implemented ECWEB, a supervised forum crawler. We reduced the forum crawling problem to a URL type recognition problem and showed how to leverage implicit navigation paths of forums. ECWEB can effectively apply learnt forum crawling knowledge on 160 unseen forums to automatically collect index URL, thread URL, and page-flipping URL training sets and learn ITF regexes from the training sets. ECWEB is indeed very effective and efficient and outperforms well.

VII. FUTURE ENHANCEMENT

In the future, we would like to discover new threads and refresh crawled threads in a timely manner. The initial results of applying a ECWEB like crawler to other social media are very promising. Also, there is plan to conduct more comprehensive experiments to further verify our approach and improve upon it.

REFERENCES

[1] Jingtian Jiang, Xinying Song, Nenghai Yu,and Chin-Yew Lin, FoCUS : Learning to Crawl Web Forums, IEEE transactions, 2013.
[2] X.Y. Song, J. Liu, Y.B. Cao, and C.-Y. Lin. Automatic Extraction ofWeb Data Records Containing User-Generated Content. Proc. 19thInt'l Conf. Information and Knowledge Management, pp. 39-48, 2010.
[3] Y. Wang, J.-M. Yang, W. Lai, R. Cai, L. Zhang, and W.-Y. Ma. ExplorinTraversal Strategy for Web Forum Crawling. Proc. 31st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp.459-466, 2008.
[4] Y. Zhai and B. Liu. Structured Data Extraction from the Web basedon Partial Tree Alignment. IEEE Trans. Knowl. Data Eng., vol. 18, no.12, pp. 1614−1628, 2006.
[5] J. Zhang, M. S. Ackerman, and L. Adamic. Expertise Networks inOnline Communities: Structure and Algorithms. Proc. 16th Int'l Conf. World Wide Web, pp. 221-230, 2007.
[6] 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.
[7] Pages in WWW Forums. Computer Engineering, vol. 33, no. 6, pp. 80-82, 2007.
[8] L. Zhang, B. Liu, S. H. Lim, and E. O'Brien-Strain. Extracting and Ranking Product Features in Opinion Documents. Proc. 23rd Int'l Conf. Computional Linguistics, pp. 1462-1470, 2010.
[9] De-duping URLs via re-write rules. Proc. 14th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 186-194, 2008. A. Dasgupta, R. Kumar, and A. Sasturkar.
[10]] 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.