Journal of Advances and Scholarly Researches in Allied Education
Vol. XV, Issue No. 4, June-2018, ISSN 2230-7540

Intelligent Web Crawler by Supervised Learning

Deepak Ranoji Naik[1]* Prof. Dr. Satish R. Todmal[2]

[1] Student, JSPM's ICOER, Wagholi, Pune

[2] Professor, JSPM's ICOER, Wagholi, Pune

Abstratct – In this paper we present Intelligent Web Crawler (IWC) a supervise and intelligent web scale forum crawler. The goal and objective of this IWC is to crawl relevant forum content from the web with minimum overhead. URL and forum threads have information content that is collected by forum crawlers. Web forum crawling problem to a URL type have been reduced to recognition problem which shows how to learn accurate and effective regular expression patterns of constant navigation paths by automatically created training sets using aggregated results from weak page type classifiers. Every forum have different layouts or styles and have different forum software packages, they always have homogeneous constant navigation paths connected by specific URL types to direct users from entry pages to thread page. Robust page type classifiers can be get from as few as five annotated forums and applied to a large set of unseen forums. To have accurate specification we have used the supervise machine learning process applied to immense set of Forum. Among the other forum crawlers, IWC gives best performance. The results show that IWC gives better performance in terms of precision and crawling time. In future, we would like to extend this crawler to other sites like Question & Answer (Q & A) sites, blog sites and other social media sites to develop as IWC as better forum crawler.

Keywords: URL; Forum Crawling; ITF Regex; URL Type; Page Type; Irobot; IWC; Crawler;

----------X----------

1. INTRODUCTION

INTERNET forums [16] called web forums are important services where users can request and exchange information with others. The World-Wide-Web is growing at a vast and rapid rate and it is finding difficult to retrieve specific information on the web. Such vast and rapid growth of the WWW causes unparallel scaling challenges for general purpose crawlers and search engines. We present a supervised web-scale forum crawler with machine learning tendency known as Intelligent Web Crawler (IWC) in this paper. IWC goal and objective is to crawl relevant forum content from the web with minimum overhead. By machine learning process it 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. IWC continuously keeps on crawling the web and finds any new web pages that have been added to the web by checking into the database, pages that have been removed from the web. It is challenging due to growing and dynamic nature of the web for traversing URLs in the web documents and to handle these URLs. We take here only 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.

Extracting or "mining" knowledge from large amounts of data and knowledge discovery in databases is known as Data Mining. Data mining discovered interesting knowledge from large amounts of data stored either in databases, data warehouses, or other information repositories. Researchers are increasingly interested in mining knowledge from them due to the richness of information in forums,. Zhai and Liu [13], Yang et al. [12], and Song et al. [11] extracted structured data from forums.

In the form of posted messages people can hold conversations on an Internet forum, or message board, is an online discussion site. ; A posted message might need to be before it becomes visible as it is approved by a moderator depending on the access level of a user or the forum setup. Each forums consist of specific set of pattern associated with them; e.g. a single conversation is called a "thread", or topic. Web Crawler the central part of the search engine which browses through the hyperlinks and stores the visited links for the future use.

The discussion forum contains hierarchical or tree-like in structure has number of sub-forums, each of which may have several topics. In forum's topic, each new discussion started is called a thread,

which is replied by many people upon wish and tag.

A forum contains tree like directory structure where the top end is "Categories" with different types of URL's and pages. Main forum is divided into sub-forums and these sub-forums can further have more sub-forums. where it is in graph structure. We used three basic message board with their own advantages and disadvantages display formats: Non-Threaded/Semi-Threaded/Fully Threaded.

Scientific discipline that explores the construction and study of algorithms that can learn from data is called Machine Learning. Such algorithms operate by building a model based on inputs and using that to make predictions or decisions, rather than following only explicitly programmed instructions.

A page and link structure in a forum is given below. The user can navigate from the entry page to thread page

There are the different paths in forum

1. entry → board → thread
2. entry → list_of_board → board → thread
3. entry → list_of_board & thread → thread
4. entry → list_of_board & thread → board → thread
5. entry → list_of_board → list_of_board & thread → thread
6. entry→ list_of_board →list_of_board & thread → board → thread

The pages between the entry page and thread page which are on a the index page. The represent implicit path is called as the EIT path. Forums exit the different layouts or styles and software packages.

EIT path is also known as the entry_index_thread page.

entry page →index page→ thread page

2. LITERATURE SURVEY

In software development process literature survey is the most important step. Before developing the project it is necessary to determine the time factor, economy and company strength. Next steps are to determine which operating system and language can be used for developing the project and required resources.

The existing system is a manual or semi- automated system. It consumes large amount of data space and time. So it is not flexible for the current environment.

Vidal et al. [25] , Guo et al [17] and Li et al [20] had work towards the method for learning regular expression patterns of URLs which guide a crawler from an entry page to target pages. By preselected sample target page, target pages were found through comparing DOM trees of pages. It is very effective but it only works for the specific site from which the sample page is drawn. The same process has to be repeated every time for a new site hence not suitable for large-scale crawling.

2.1 Type of Crawler:

2.1.1 Generic Web Crawler

Breadth-first traversal strategy, are usually ineffective and inefficient for forum crawling which is adopted by Generic Web Crawler (Brin and Page, 1998). which is mainly due to two non crawler-friendly characteristics of forums (Wang, et. al., 2008), (Zhai and Liu, 2006) i.e. duplicate links and uninformative pages and second is page-flipping links. Generic crawlers ignore the relationships between such pages and process each page individually and the relationships must be preserved while crawling to facilitate downstream tasks such as page wrapping and content indexing (Yang, et. al., 2009).

It is Automatic traversal of web to collect all the useful informative pages, effectively and efficiently gather information about link structure interconnecting the informative pages. Web application designed to manage user created content and store in Database. It is online discussion area where anyone can discuss their favorite topics.

It Pre-samples few pages to discover the repetitive regions and group pre-sampled pages are clusters based on their repetitive regions where each cluster can be considered a vertex in the sitemap.

2.1.2 Focused Crawler:

A Focused Crawler is designed to crawl Web pages relevant to a pre-defined topic (Jingtian et. al., 2013). There are several Focused Crawlers available each using different techniques.

A Semantic Focused Crawler (Koppula, et. al., 2010) is a Focused Crawler which makes use of Semantic Web technologies for performing the crawling.

a) An Ontology-based Semantic Focused Crawler links Web documents with related

Ontology concepts for the purpose of categorizing them by making use of ontology to analyze the semantic similarity between URLs of Web pages and topics. Limitation: Most of these crawlers fetch the surrounding texts of URLs as the descriptive texts of the URLs and compute the similarity between the URLs and ontology concepts based on these texts and script.

b) A Metadata Abstraction based Semantic Focused Crawler extracts meaningful information or metadata from relevant Web pages which annotates the metadata with Ontology Mark-up Languages.

A Focused Crawler based on Link Structure and Content Similarity makes use of a combination of link-structure and similarity of contents between the Web pages for performing the crawling (Wang, et. al., 2008),. This crawler starts with an initial 'Seed'. If the initial Seed page does not relate to the domain, then the number of related pages will be very less in the beginning stages. This will affect the overall efficiency of the crawler.

2.1.3 Irobot

Web Forums is intelligently crawl by (Wang, et. al., 2008) enough to understand structure of forums before selecting traversal path and search into the graph. Irobot automatically learn a forum with minimum human intervention by sampling pages, clustering them, selecting informative clusters via an in formativeness measure, and finding a traversal path by a spanning tree algorithm which requires human inspection. Its look for important page and important links. Wang et al. [14] proposed an algorithm to address the traversal path selection problem. High effectiveness and coverage is achieved by Irobot by identifying and following skeleton links and page-flipping links.

3. TERMINOLOGY

3.1 Collection of Forum

Index URLs, thread URLs, and page-flipping URLs have specific URL patterns in forum. By learning patterns of index URLs, thread URLs, and page-flipping URLs and adopting a simple URL string with de-duplication technique which is performed by supervise machine learning process. Repetition and duplication is avoided by IWC without duplicating detection Page Type.

3.2 Page Types

3.2.1 Entry Page: Entry Page is the homepage of a forum, which contains a list of boards and is also the lowest common ancestor of all threads.

3.2.2 Index Page: Index Page is a page of a board in a forum, which usually contains a table-like structure; each row in it contains information of a board or a thread. Collect list-of- board page, list-of-board and thread page, and board page are all index pages.

3.2.3 Thread Page: Thread Page **is** a page of a thread in a forum that contains a list of posts with user generated content belonging to the same discussion.

3.2.4 Other Page: A page that is not an entry page, index page, or thread page.

3.3 Type of URL

There are four types of URL.

3.3.1 Index URL: A URL that is on an entry page or index page and points to an index page is called Index URL. Its anchor text shows the title of its destination board.

3.3.1 Thread URL: A URL that is on an index page and points to a thread page is called Thread URL. Its anchor text is the title of its destination thread.

3.3.2 Page-flipping URL: A URL that leads users to another page of the same board or the same thread is called Page-flipping URL. Correctly dealing with page-flipping URLs enables a crawler to download all threads in a large board or all posts in a long thread.

3.3.3 Other URL: A URL that is not an index URL, thread URL, or page-flipping URL.

3.3.4 EIT Path: An entry-index-thread path is a navigation path from an entry page through a sequence of index pages to thread pages.

3.3.5 ITF Regex: An index-thread-page-flipping regex is a regular expression that can be used to recognize index, thread, or page-flipping URLs. IWC learns ITF regex and applies directly in online crawling. The learned ITF regexes are site and forum specific.

4. FEASIBILITY STUDY

The feasibility study of the project is analysed in this phase of development and in terms of cost estimation for business purposed. It is to be carried out during system analysis and ensure that the proposed system is not a burden. Major requirements for the system are three key considerations:

- Economical Feasibility

- Technical Feasibility
- Social Feasibility

4.1 Economical- Feasibility

Checks for the economic impact that the system will have on the organization. Due to limited fund for the research and development of the system the expenditures must be justified. Our current project is in budget as most of the technologies are freely available. Only the customized products had to be purchased.

4.2 Technical Feasibility

Checks for the technical feasibility i.e. the technical requirements of the system. The system developed by organization must not have a high demand on the available technical resources as it will lead to high demands being placed on the client. Our current project has a modest requirement, as only minimal or null changes are required for implementing this system.

4.3 Social Feasibility

Checks the level of acceptance of the system by the user as it includes the process of training the user to use the system efficiently and conveniently. The user must accept it as a necessity but not get threatened. The user must be familiar and educated about the system and must have high level of confidence so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

In this project, user is a Netizen and will search on a web based forum to search the relevant content and will be displayed in that page.

5. SUPERVISE MACHINE LEARNING PROCESS

Machine learning is a sub field of computer science and statistics which has strong ties to artificial intelligence and optimization, which deliver methods, theory and application domains to the field.

In a range of computing tasks where designing and programming explicit, rule-based algorithms is infeasible Machine Learning is employed. Ex: applications include spam filtering, optical character recognition (OCR), search engines and computer vision. Machine learning is sometimes compared with data mining, although that focuses more on exploratory data analysis. Machine learning and pattern recognition "can be viewed as two facets of the same field.

Input data is a training data which has a known label or result such as spam/not-spam or a stock price at a time. A model is prepared through a training process where it is required to make predictions and is corrected when those predictions are wrong or incorrect where training process continues until the model achieves a desired level of accuracy on the training data. Ex: problems are classification as well as regression. Second ex: algorithms are Logistic Regression and the Back Propagation Neural Network in a system.

5.1 Steps in Supervise Learning

A known set of input data and known responses to the data is taken by Supervised learning (machine learning) which seeks to build a predictor model that generates reasonable predictions for the response to new data.

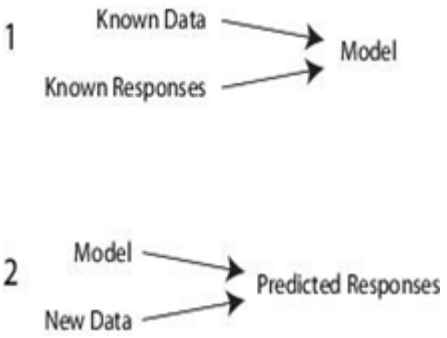

Fig (a) Supervise Learning

Example: you want to predict if someone will have a heart attack within a year. You have a set of previous data including age, weight, height, blood pressure, etc. You know if the previous person had heart attacks within a year of their data measurements. So the problem combines all the existing data into a model which will predict whether a new person will have a heart attack within a year.

5.1 Supervised learning splits into two broad categories:

Classification for responses that can have just a few known values, such as 'true' or 'false' which apply to nominal, not ordinal response values. Regressions for responses are a real number, such as miles per gallon for a particular car. You can have problem in deciding whether you have a classification problem or a regression problem. In this case, create a regression model first, because they are often more computationally efficient.

Most use the same basic workflow for obtaining a predictor model while there are many Statistics Toolbox algorithms for supervised learning,.

(Detailed instruction on the steps for ensemble learning is in Framework for Ensemble Learning.)

The steps for supervised learning are:

1. Prepare Data
2. Choose an Algorithm
3. Fit a Model
4. Choose a Validation Method
5. Examine Fit and Update Until Satisfied
6. Use Fitted Model for Predictions

SVM prediction speed and memory usages are best if there are few support vectors, but can be worst if there are many support vectors. It become difficult to interpret how SVM classifies data, though the default linear scheme is easy to interpret when you use a kernel function

Naive Bayes speed and memory usage are best for simple distributions, but can be worst for kernel distributions and large data sets.

Nearest Neighbor usually has best predictions in low dimensions, but can have worst predictions in high dimensions. Nearest Neighbor does not perform any fitting for linear search. For kd-trees, Nearest Neighbor does perform fitting which have either continuous or categorical predictors, but not both.

Discriminant Analysis is accurate when the modeling assumptions are satisfied. (Multivariate normal m by class) otherwise predictive accuracy varies.

5.1.1 Framework for Ensemble Learning

You have several methods for melding results from many poor or weak learners into one high-quality ensemble predictor. These methods follow closely the same syntax, so you can try different methods with minor changes in your commands.

Create an ensemble with the fitensemble function. Its syntax is ens = fitensemble(X, Y, model, numberens, learners)

1. X is the matrix of data. Each row contains one observation, and each column contains one predictor variable.
2. Y is the vector of responses, with the same number of observations as the rows in X.
3. Model is a string naming the type of ensemble.
4. Numberens is the number of weak learners in ens from each element of learners. So the number of elements in ens is numberens times the number of elements in learners.
5. Learners is either a string naming a weak learner, a weak learner template, or a cell array of such templates.

Pictorially, here is the information you need to create an ensemble:

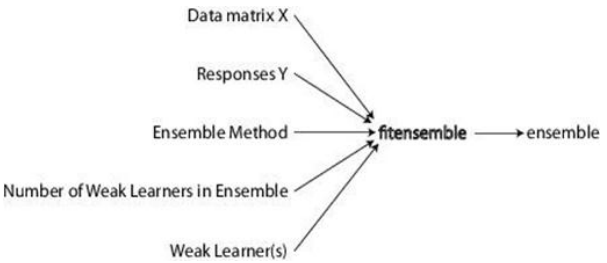

Fig (b) Ensemble Learning

6. STUDIES AND FINDINGS

6.1 Read URL

We are concentrating on IWC ontology which search and finds for the relevant web pages or contents from forum based on the keyword we give to it by forming a hierarchy of links. It searches the web information on the particular web page for a particular keyword, which we give as, input. It goes for the link through seed URL and switches to that link and find another link on that web page but it ,must match with the keyword, it will do like that until it reach the limit that we had set. It is not sure that it will find the number of links that we set before and can shows that the web page is not having any further link for that particular keyword. While fetching the links the user profiles should make sure that it must fetch only the unique links, i.e. it must not revisit the same link again and again or avoid duplication. Give one text file as input and run the three pattern matching algorithm finally when we finished with the links.

6.2 Pattern Recognition

Pattern recognition means only text. For syntax analysis Pattern matching is used. When we compare pattern matching with regular expressions then we will find that patterns are more powerful, but slow in matching. A pattern is nothing but 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 where spaces and tabs can be used to separate keywords. Text or alphabet mining is one of the important steps of knowledge discovery process which extracts hidden information from non-structured or semi-structured

data. Mining 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 or repetitive. The whole knowledge mining process of mining, extraction and integration of useful data, information and knowledge from the web page content is done by Web text mining. By applying pattern recognition on the web information in this manner, while retrieval it will give me the links related to the keyword and will read the web pages that are extracted and retrieve from the links. While it will read the web page it will extract only the content. The contents are only the text that is available on the web page or forum page and must not include images, tags, and buttons. It must be stored in some file and must not include any HTML tags.

6.3 Identification Process

The 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 the data. It's omits 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 store as database.

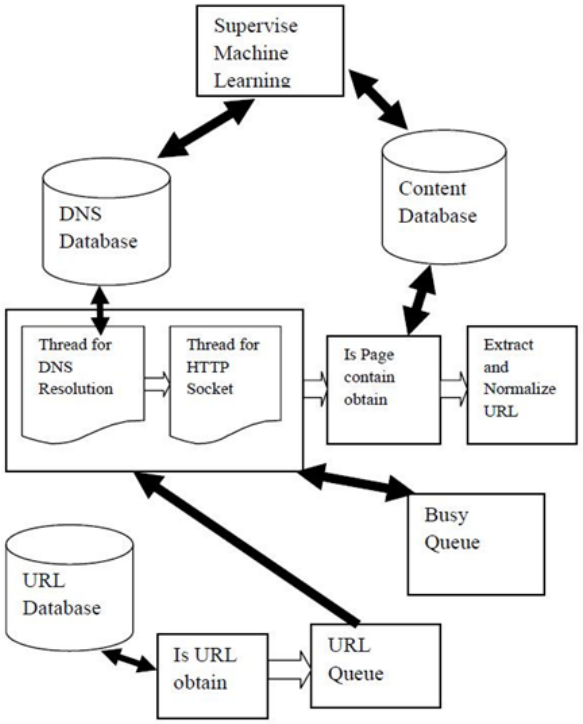

Fig (c) Architecture of Web Crawler by using Machine Learning

6.4 Downloading Process

URL link of users needed after completion of all process the downloading will started by starting downloading requesting. It will download the relevant link according to users request after three checking process only. It will be working efficiently to users while the requested link will retrieve.

6.5 Index URL and Thread URL Training Sets

It is a URL that is on an entry or index page and its destination page is another index page and its anchor text is the board title of its destination page. A thread URL is a URL that is on an index page; its destination page is a thread page; its anchor text is the thread title of its destination page. Difference between index URLs and thread URLs is the type of their destination pages. Therefore, it is necessary to decide the page type of a destination page. The index pages and thread pages each have their own typical layouts i.e. index page has many narrow records, relatively long anchor text, and short plain text; while a thread page has a few large records each post has a very long text block and relatively short anchor text. An index page or a thread page always has a timestamp field in each record, but the timestamp order in the two types of pages are reversed i.e. the timestamps are typically in descending order in an index page while they are in ascending order in a thread page.

The Index URL and Thread URL detection algorithm is given below.

1. Enter data
2. To collect all URL groups and longest link text length
3. Select URL group
4. IF the pages are not Index or Thread page then discarded.

6.6 Page-Flipping URL Training Set

It points to index pages or thread pages but are different from index URLs or thread URLs.

The "connectivity" metric distinguish page-flipping URLs from other loop-back URLs but it only works well on the "grouped" page-flipping URLs, i.e. more than one page-flipping URL in one page.

In particular, the grouped page-flipping URLs have the following properties:

1. Their anchor text is either a sequence of digits such as 1, 2, 3, or special text such as "last."

2. They appear at the same location on the DOM tree of their source page and the DOM trees of their destination pages.

3. Destination pages have similar layout with their source pages. Tree similarity is used to determine whether the layouts of two pages are similar or not.

4. The single page-flipping URLs appearing in their source pages and their destination pages contains same anchor text but different URL strings.

Our page-flipping URL detection module works on above properties. The detail is shown in Fig. 3. Lines 1-11 first tries to detect the "group" page-flipping URLs; if it failed, lines 13-20 will enumerate all the outgoing URLs to detect the single page-flipping URLs; and line 23 set its URL type to page-flipping URL.

In our experiment over 120 forum sites (10 pages each of index and thread page), our method achieved 95 percent recall and 99 percent precision. We apply this method to both index pages and thread pages; they found page-flipping URLs are saved as training examples.

The Page Flipping URL detection algorithm is given below.

1. To detect the group page-flipping URLs if it fails.

2. It enumerates all the outgoing URLs to detect the single page-flipping URLs.

3. Set its URL type to page-flipping URL.

6.7 Evaluation of Entry URL Discovery

We assume that in forum crawling an entry URL is given. But finding forum entry URL has more value. For each forum in the test set, sampled page is fed to the module and it is manually checked if the output was indeed its entry page. In order to see whether low standard deviation indicates that it is not sensitive to sample pages. Two main failure cases: 1) forums are no longer in operation and 2) JavaScript generated URLs which we do not handle currently. Entry URL Discovery algorithm:

1. URL check in all forum If keyword found ,the path from URL host.

2. Every page in a forum site contains a link to lead users back to its entry page.URL is detected as an index URL

3. An entry page has most indexes URLs Since it leads users to all forum threads.

6.7 Evaluation of Online Crawling

IWC is efficient in learning ITF regexes and is effective in detection of index URL, thread URL, page-flipping URL, and forum entry URL and thus we compare IWC with other existing methods in terms of effectiveness and coverage.

6.8 ITF Regexes Learning

Goal of URL training sets is to construct automatically sets of highly precise index URL, thread URL, and page-flipping URL strings for ITF regexes learning for supervise machine learning.

IWC first learns a set of ITF regexes and performs online crawling using a breadth-first strategy. Every time it pushes the entry URL into a URL queue; next it fetches a URL from the URL queue and downloads its page; and then it pushes the outgoing URLs which are matched with any learned regex into the URL queue. IWC repeats the above step until the URL queue is empty or other conditions are satisfied and no need to group outgoing URLs, classify pages, detect page-flipping URLs, or learn regexes again for that forum.

Generally all search engines have highly optimized crawling system, but working and details of documentation of this system are usually with their owner. It is easy and simple to create a crawler that would work slowly and download few pages per second for a short period of time but a big challenge to build the perfect system design, I/O, network efficiency, robustness and manageability. Crawler module is the module on which search engine depends the most among the different module because it helps to provide the best possible results in a search engine.

7. ANALYSIS OR TEST SPECIFICATION

Testing detect software failures so that defects may be discovered and corrected and determines the correctness of software under the assumption of some specific hypotheses. It is investigation procedure of quality information. It validates and verify so that the requirement is met. It is document describing the scope, approach, resources and schedule of intended activities. The objective is to find and report as many bugs as possible to improve the integrity of the system. Although exhaustive testing is not possible, a broad range of tests is exercised to achieve the goal of bug free and accurate results in software. The software testing has been done for all components in every module of software. The input and output of each module are tested to be accurate and valid for giving particular datasets. The results of modules are compared to existing protocols results.

7.1 Intelligent Web Crawler

Earlier crawling system learn regular expression patterns of URLs that lead a crawler from an entry page to target pages by comparing DOM trees of pages with a pre-selected sample target page. It is very effective and works only for the specific site from which the sample page is drawn. Every time for a new site the same process has to be repeated Therefore, it is not suitable to large- scale crawling. The generic crawler started from the entry URL and a randomly selected non entry URL, respectively. When no more pages could be retrieved it stopped.

A crawler following only index URLs, thread URLs, and page-flipping URLs will achieve almost 100 percent effectiveness. When starting from entry URL, all coverage are near about100 percent. When starting from a nonentry URL, coverages decreased significantly. an entry URL is crucial for forum crawling. We propsed IWC as smart web crawler which learns URL patterns across multiple sites and automatically finds a forum's entry page given a page from the forum as well as URL patterns instead of URL locations to discover new URLs and there is no need to classify new pages in crawling and would not be affected by a change in page structures. The respective results from Google, Bing and IWC demonstrated that the EIT paths and URL patterns are more robust and promising than the traversal path and URL location feature in Google and other web crawler. IWC avoids duplicates URL without duplicate detection by learning patterns of index URLs, thread URLs, and page-flipping URLs and adopting a simple URL string de-duplication technique for example a string hashset.

8. RESULTS AND DISCUSSIONS.

Different logged users are tested. The following figures below show the login part and keyword search part. It also shows the analysis of the different user.

8.1 Learning efficiency comparison.

We evaluated the efficiency of IWC and other methods in terms of the number of pages crawled and the time spent during the learning phase and evaluation.

We evaluated and calculated how many pages are necessary to get satisfactory results for these methods. We evaluated results under the metric of average coverage over the two forums and limit the methods to sample at most N pages, where N varies from 10 to 3,500. Then we let the methods crawl forums using the learned pattern and knowledge. Fig 7 shows the no of pages and internal link.

Fig. 8 shows the average coverage based on different numbers of sampled pages. We can see that, IWC needs very less pages to achieve a stable performance as compare to other forum crawler. Fig 10 shows that it takes less step as compare to other forum crawler.

To estimate the time spent on a forum during the learning phase, we ran these systems on a machine with core(TM) i3 CPU, 2.40 GHz CPUs, 4 GB memory, and 64-bit Ubuntu 18.04 OS. The maximum time that IWC spent on a forum was 5.504 seconds, the minimum time was 1.228 seconds, and the average was 3.366 seconds. If you search for the first time the keyword IWC will take more time but next time the time period drop down drastically as shown in fig 11. According to our experience, even a skilled person would spend about 1,200 seconds on average to write URL rules for a forum. In contrast other structure-driven crawler used 2.273 seconds on average. Compared to Google , IWC learns the EIT path and ITF regexes directly and precisely. Therefore, IWC was not affected by noisy pages and performed better. This indicates that given fixed bandwidth and storage, IWC can fetch much more valuable content than Bing.

According to Fig. 9, IWC had better coverage than the structure-driven crawler, Bing and Google. The average coverage of IWC was 99 percent, compared to structure-driven crawler's 92 percent. From the results, we can conclude that the structure-driven crawler with small domain adaptation cannot be an effective forum crawler. In contrast, all the threads in forums are almost crawled by IWC as it learned EIT path and ITF regexes directly. IWC found all boards and threads regardless their locations in online crawling. The coverage results show that our methods of index/thread URL and page-flipping URL detection are very effective.

System average calculates the average effectiveness or coverage over all pages from all forums and within a forum as shown in fig 8 and 9. Question thread pages or blog post pages do not contain page-flipping URLs. Thus, we do not need to detect page-flipping URLs in these pages. Without page-flipping URLs, a crawler could only get the first page of each board, and misses all threads in the following pages. Thus, the coverage might be very low. We proposed solving immense scale forum crawling problem as a URL type recognition problem by identifying the EIT path through learning the ITF regexes. IWC achieved nearly 100 percent effectiveness and coverage on all sites.

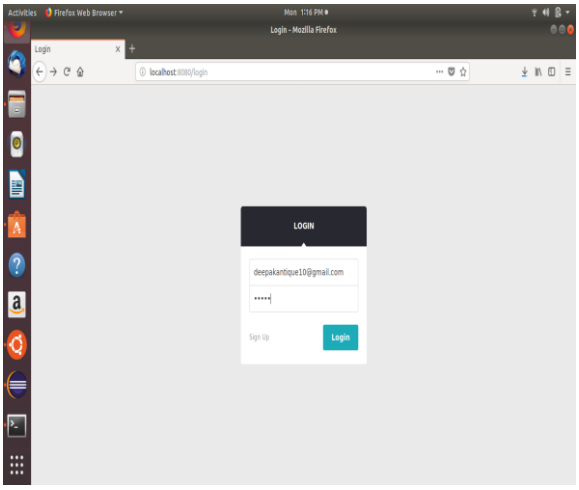

Fig (d) Login Window

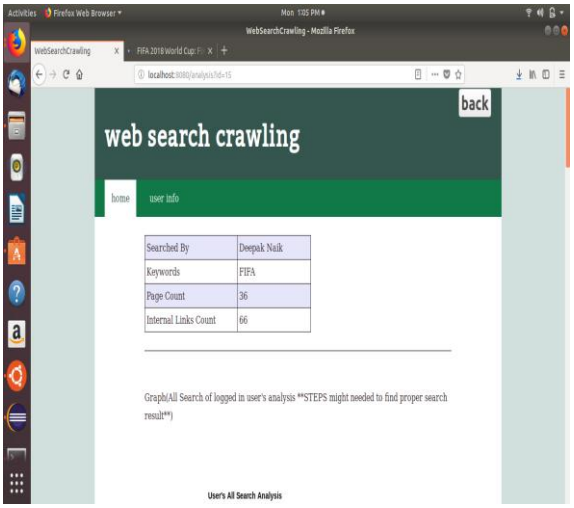

Fig (g) Page count and Internal Link Count

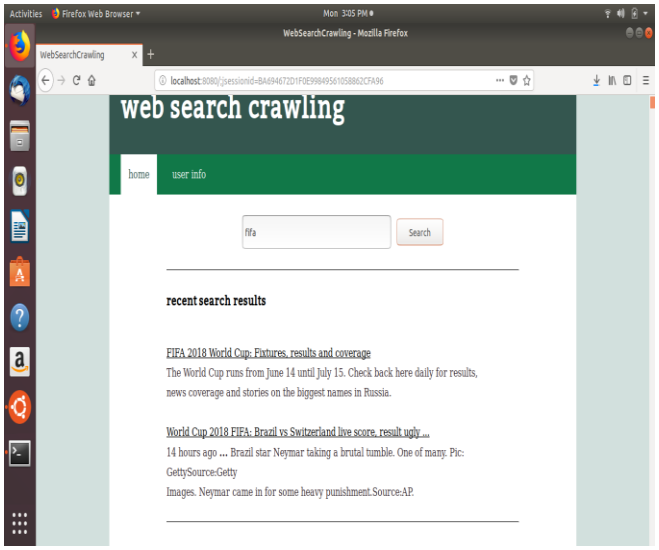

Fig (e) Searching of keyword by user

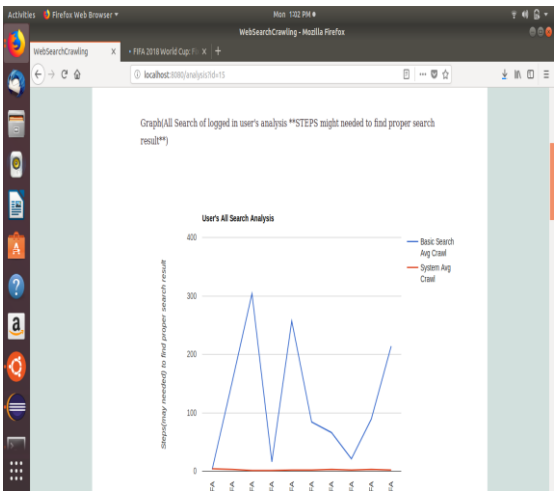

Fig (h) System Crawl Analysis

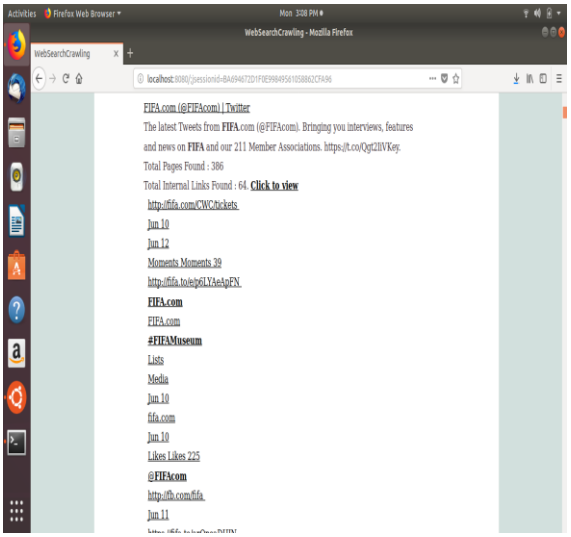

Fig (f) URL Link along with no of pages and internal links

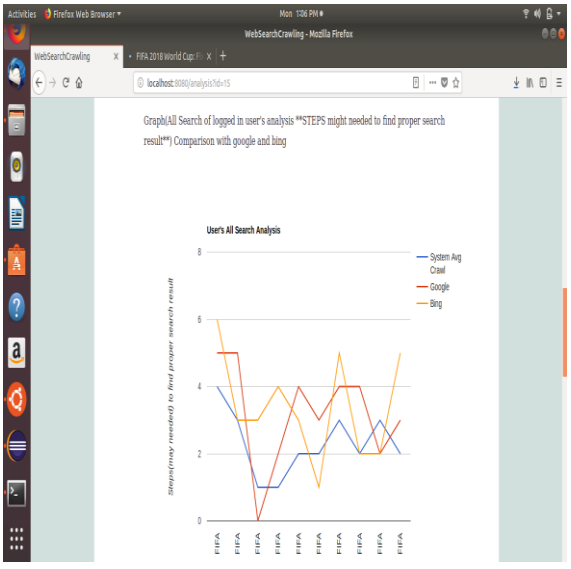

Fig (i) Comparison between different Web Forum and IWC

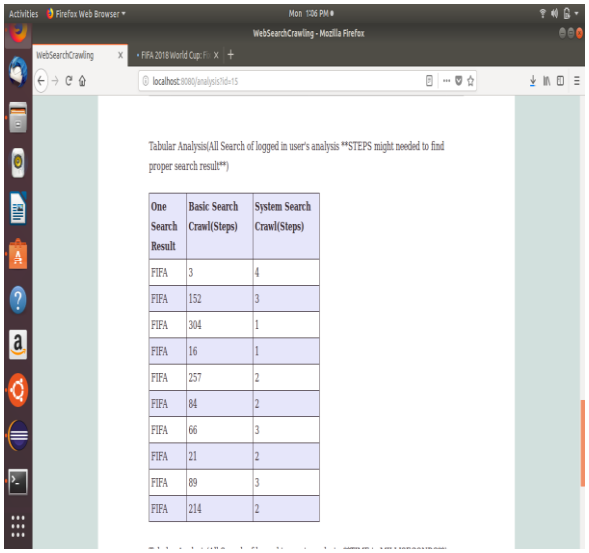

Fig (j) Testing of logged user in terms of steps

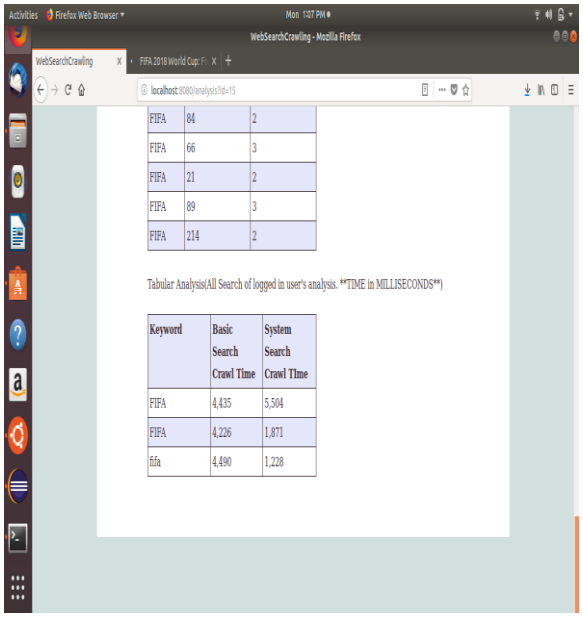

Fig (k) Testing of logged user in terms of time

9. DETAILED RESULTS

Results shows that IWC crawls forum relatively fast as compare to other crawler. It also always points the current URL the user has visited. IWC achieved better performance than the structure-driven crawler, Bing and Google. IWC achieved nearly 100 percent effectiveness and coverage on all sites. The lowest effectiveness is 99.92 percent and the lowest coverage is 99.10 percent. These results make sure that IWC can be applied to crawl cQA sites.

10. CONCLUSION

We are working on IWC which crawl the forum data automatically and clean up the unwanted data and allocate that space to new queries posted by the user. IWC crawler based on the keyword searches the relevant web pages and forms a hierarchy of links for the crawler on the particular web page for a particular keyword, which we provide as input. By keyword matching crawler will search for the link on that seed URL and will switch to that link and find another link on that web page. It will process until it reach the limit that we set based upon the machine learning process. Partial-match, of Knutt-Morris-Pratt method identifies the bad URL in a website and number of character present in a web page. IWC identifies type of protocol used for the web page. And retrieve the web pages. We apply pattern recognition over text for correct navigation. Pattern symbolizes check text only i.e. what quantity text is available on web page. Learned patterns are effective and the resulting crawler is efficient as per the result. In terms of effectiveness and coverage IWC outperforms the other crawlers. Among the other forum crawlers, IWC gives best performance. The results show that IWC gives better performance in terms of precision and crawling time. A detailed study of IWC disclosed some drawbacks in it. In future, we would like to extend this crawler to other sites like Question & Answer (Q & A) sites, blog sites and other social media sites also.

REFERENCES

C. Gao, L. Wang, C.-Y. Lin, and Y.-I. Song (2008). "Finding Question- Answer Pairs from Online Forums," Proc. 31st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 467-474.

ForumMatrix, http://www.forummatrix.org/index.php, 2012.

G.S. Manku, A. Jain, and A.D. Sarma (2007). "Detecting Near-Duplicates for Web Crawling," Proc. 16th Int'l Conf.World Wide Web, pp. 141-150.

H.S. Koppula, K.P. Leela, A. Agarwal, K.P. Chitrapura, S. Garg, and A. Sasturkar (2010). "Learning URL Patterns for Webpage De- Duplication," Proc. Third ACM Conf. Web Search and Data Mining, pp. 381-390.

Internet Forum, http://en.wikipedia.org/wiki/Internet_forum, 2012.

J.-M. Yang, R. Cai, Y. Wang, J. Zhu, L. Zhang, and W.-Y. Ma (2009). "Incorporating Site-Level Knowledge to Extract Structured Data from Web Forums," Proc. 18th Int'l Conf. World Wide Web, pp. 181- 190.

Jingtian Jiang, Xinying Song, Nenghai Yu, and Chin-Yew Lin (2013) "FoCUS: Learning to Crawl Web Forums "IEEE, Transactions on Knowledge and Data Engineering.

K. Li, X.Q. Cheng, Y. Guo, and K. Zhang (2007). "Crawling Dynamic Web Pages in WWW Forums," ComputerEng., vol. 33, no. 6, pp. 80-82.

M. Henzinger (2006). "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.

N. Glance, M. Hurst, K. Nigam, M. Siegler, R. Stockton, and T. Tomokiyo (2005). "Deriving Marketing Intelligence from Online Discussion," Proc. 11th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 419-428.

nofollow, http://en.wikipedia.org/wiki/Nofollow, 2012.

R. Cai, J.-M. Yang, W. Lai, Y. Wang, and L. Zhang (2008). "iRobot: An Intelligent Crawler for Web Forums," Proc. 17th Int'l Conf. World Wide Web, pp. 447-456.

S. Brin and L. Page (1998). "The Anatomy of a Large-Scale Hypertextual Web Search Engine." Computer Networksand ISDN Systems, vol. 30, nos. 1-7, pp. 107-117.

The Web Robots Pages," http://www.robotstxt.org/, 2012.

U. Schonfeld and N. Shivakumar (2009). "Sitemaps: Above and Beyond the Crawl of Duty," Proc. 18th Int'l Conf.World Wide Web, pp. 991-1000.

WeblogMatrix," http://www.weblogmatrix.org/, 2012.

X.Y. Song, J. Liu, Y.B. Cao, and C.-Y. Lin (2010). "Automatic Extraction of Web Data Records Containing User - Generated Content," Proc. 19th Int'l Conf Information and Knowledge Management, pp. 39-48.

Y. Guo, K. Li, K. Zhang, and G. Zhang (2006). "Board Forum Crawling: A Web Crawling Method for Web Forum," Proc. IEEE/WIC/ACM Int'l Conf. Web Intelligence, pp. 475-478.

Y. Wang, J.-M. Yang, W. Lai, R. Cai, L. Zhang, and W.-Y. Ma (2008). "Exploring Traversal Strategy for Web Forum Crawling," Proc. 31st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 459-466.

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

Corresponding Author

Deepak Ranoji Naik*

Student, JSPM's ICOER, Wagholi, Pune

E-Mail – deepakantique10@gmail.com