ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 2, January 2017

Keen Crawler: A Novel Two-Stage Crawler Framework for Searching the Hidden-Web Resources

Gandhavalla Dinesh Kumar [1], Dr. B. Venkata Seshu Kumari[2]
M.Tech Student, Information Technology, VNRVJIET, Hyderabad, India [1]
Assoc.Professor, Information Technology, VNRVJIET, Hyderabad, India [2]

Abstract: The exploitation of internet is growing second to second extensively by each & every individual. This lead to the growth of data in a bulky manner, but only the top surface data is frequently crawled which resulted in the accumulation of unexposed web resources. There is huge information in internet which is not utilized. These deep unexposed data cannot be crawled using a web search engine as this type of data's are not enrolled with any web search engine databases. This paper suggestsa crawler framework for searching deep unexposed web resources. It consists of 2 stages in the framework namely 1. Site Discovering and 2. Site Analysing. This framework produces efficient search results for unexposed web resources employing existing search engines. A search string is given as keyword to locate and explore the information present in deep unexposed web resources.

Keywords: Deep web, crawling, web-mining, feature selection, ranking

I.INTRODUCTION

Internet is an endless reserve of information. But the information utilized by any individual is very less & it is dependent on most retrieved web pages. Web pages were given ranking by search engine's provisionalto the most searched terms. This lead to the generation of deep unexposed web resources. As these content have low ranking search engine assumes it as less relevant pages but relevant information will be present where the spider fails in crawling them. Web spidering is a mechanism of retrieving the significant information from internet. WebCrawler is the internet's backbone & the program which performs these operations are generally known as spiders or bot.

Internet is dynamic in nature & it keeps on growing every second. It contains various types of data which makes it difficult in retrieving for the crawlers. The data in internet are of searchable forms & non-searchable forms. This was the problem for slow search and retrieving process by a crawler. This difficulty can addressed, as previous papers suggested two categories like Comprehensive Crawler & Specialized Crawler. Comprehensive Crawlers are nothing but generic crawlers those can retrieve all searchable forms but cannot point of a distinct topic. Specialized Crawlers are focused crawlers by which spidering can be done on a distinct topic.

There are specialized crawlers like form-specialized crawlers and ACUE (Adaptive crawler for unexposed entries) which were able to examine all the content from online directories on a distinct topic. These specialized crawlers are designed in such a way, that they can crawl specific type of information using different classifiers like page classifier, link classifier & form classifier etc., . ACUE is the continuation of Form-specialized crawler with some extra features like adaptive link classifier and form refiners.

Web-Search engine's utilize crawlers to arrive at destination & extract the information. Crawlers generally use the hyperlinks available in HTML pages to locate and index the content by using protocols like virtual port numbers. The main problem with the deepwebis those are neither indexed by any of the search engine. The size of deepweb keeps on growing every second & it's almost in petabytes. Estimations providedbrief view about deepweb is almost 600 times superior than normal web surface.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 2, January 2017

The archetype proposed in [1] is advanced in this paper to achieve &explore the unexplored hidden web. The hidden net is considered as the darknet of internet which were not registered into any of the search engine's databases. This paper runs a crawler framework which can retrieve the darknet of internet & thereby resulting wide coverage area for the internet. Usually the focused crawlers are used for penetrating the darknet on a definite topic, but for the darknet there might be countless form data which clues to diverse locations thereby forming a deep web. To mitigate this problem, this paper practices a form-focused crawler that helps in accomplishing better crawling results. The keen crawler involves in 2 phases, Site Discovering & Site Analysis. The Site Discovering phase follows focused crawler & perform crawling procedure that results in wide handling of sites. The Site Analysis phase performs explorations for web forms inside a site. Along with this, the Site Discovering phase also employs a method called as *In-Depth Searching* for fetching inside the deep site when the amount of links to fetch is below than assured threshold value. A cumulative site prioritize procedure is used in this paper to acquire searchable forms in the site & Out of site of links. The Adaptive site learner and Adaptive link learner procedures are used for performing programmed feature selections that are used to figure out link rankers. One more contribution of this paper is a web map generator which extracts all the images of .png .gif & .jpeg formats available in a particular site.

The remaining paper is defined as follows. The paper gives clear background information& related word in Section II while Section IIIoffers the Design of the architecture & Analysis phase. Section IVoffersresults & Section V concludes the paper.

II. RELATED WORK

WC[1] is a process of surfinginternet pages and fetching the accurate pages based on search queries. WC are mainly comprising of different types like GC[2]& FC[3] etc., WC are also called as Spiders or Bots. WC have the capability of handling hyperlinks and HTML codes. These bots search the internet's database located on the index of web-directories and links available in that. This feature of WC made them weak as the hackers and intruders were growing day by day due to the vast coverage of internet. They intruders provided false hyperlinks and made the WC to deviate their path with resulted in fatal losses.

Then after the incidents Search engine companies hail from an idea to handle these type of trapdoors & malwares. They added a notepad file to every spider or bot which crawls internet databases. This file contains all the list of possible traps that might happened so that a spider can match the crawling Url queue with the list present in the text file and maintain its path. This text files are generally called robots file.

The Architectural flow of WC is explained in [2], which describes the progression of crawling in following steps:

1. Consider & Begin with the Base Url or Url's.
2. Add the Url's to Bound.
3. Select the Url against the Bound.
4. Retrieve the web-pages associated to selected Url.
5. Extract the all the Url's from the retrieved web-page.
6. Add all unexplored Url's to the Bound.
7. Repeat from step3 until the Bound becomes empty.

GC are generally developed for exploring the deepweb data and vast internet coverage but they fail in efficiency as they consume huge quantity of resources. For this motive, FC originated these are topic specific straight away crawl the centre pages and reduce the data consumption. There are other Crawlers like DC[4] which is used for spontaneousdetection of query interfaces was designed in MetaQuerier [3][4].It works by fetching the root pages from the server database grounded on the IP address.

There are different sorts of FC's like FFC [5] &ACHE [6]. FFC[5]are developed with variousclassifiers such as link,page,and form. The main motto of FFC is to crawl all the forms from web based on specific topic. ACHE[6] is the enriched version of FFC which consists of some additional features like form-filtering & ALL[7].

Soumen et al. proposed BFC[8] that uses page classifier to organise the search. The page classifier in BFC determines whether they standas topic specific or not & provides the links with priority. Despite of that, an improvement to BFC was proposed in [9] which uses additional classifier to choose the supreme prioritized links.

Distant from the crawling techniques discussed above *Keen Crawler* is DS[9] Crawler which houses a two-stage framework to explore the deep unexposed web resources. In its 1st phase it drains out all the unrelated websites &distinguishes the searchable- forms in its second stage. In its 2nd stage it starts in-depth searching if at all there are less links than the inception value of links to be fetched. This stage is considered as deepweb analysis.

III. DESIGN & ANALYSIS

To explore the deep hidden unexposed databases, *Keen Crawler* a 2-stage framework is developed in the paper. Keen Crawler is a domain-specific crawler which is nothing but it

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 2, January 2017

starts the crawling course from a base site or seed site given by user of some specific domain. Keen crawler is framework entailing of 2 stages namely 1.Site Discovering & 2. Site Analysis. As the internet consists of huge growing databases every second to second some data or information is not utilized because of its ranking and hidden nature. These databases has very less coverage in internet database content range, generally these are not registered into any of the web-servers of the search engines. So general crawlers or WC are not capable of crawling these hidden databases because they crawl the most used data i.e., like top layers or top surface data are crawled most of the time which makes these hidden web-resources growing into zeta bytes. This can be avoided by deep crawling or in-depth crawling of web-server databases.

[1]*Web Crawler*
[2]*Generic Crawler*
[3]*Form Crawler*
[4]*Database Crawler*

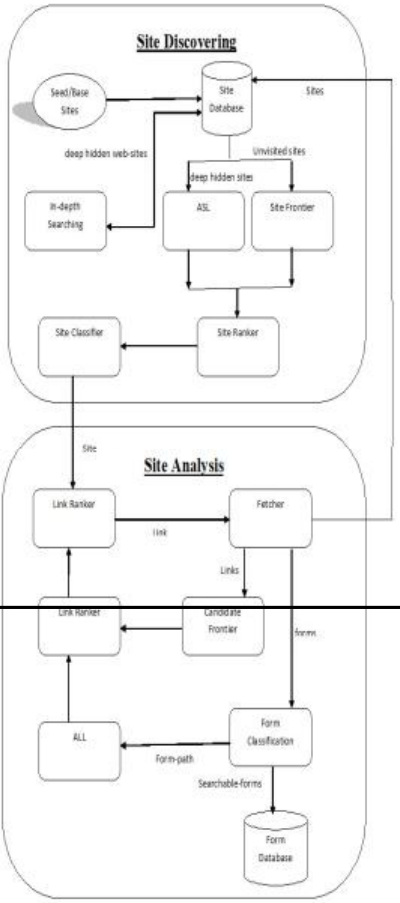

Figure 1. Architecture of *Keen Crawler*

A.Site-Discovering:

Site-Discovering is the 1st stage of the Keen Crawler. In this stage crawler is fed with an URL as input which can be called as Seed Site or Base Site. The Crawler recovers all corresponding links to that base site or seed site with the help of GC. It fetches the most significant data by selecting aFeature space thereby forming a Site Database. The links in sites database are classified into deep hidden sites and unvisited sites. The deep hidden sites are given to ASL[10] to process them. The Unexplored sites are set to Site Frontier as input & the url's pertaining to those unvisited sites will be stored in the form of a queue.

[5]*Form Focused Crawler*
[6]*Adaptive Crawler for Hidden Entries*
[7] *Adaptive Link Learner*
[8] *Best-first Focused crawlers*
[9] *Domain-Specific*

To find out the extreme related site links & Out of site links the In-Depth Searching & Cumulative Site Prioritize algorithms are used. Provisional to the yield attained from ASL & Site Frontier, site ranker nominates a rank based on the relevance. The Site Classifier catalogues the relevant sites based on the site rank given by site ranker; the classifier used in the paper is Naive Bayes. The Site Discovering phase follows focused crawler & perform crawling procedure that results in wide handling of sites.

B. In-Depth Searching Algorithm:

input: Base site and collected deep websites

output: related sites

while# *of candidate sites less than a threshold* **do**

// choose a deep website

site= getDeepWebSite(siteRecord,BaseSites)

outcomePage= In-DepthSearch(*site*)

links= mineTheLinks(*outcomeP age*)

foreach*link in links* **do**

page= takePage(*link*)

Related= order (*page*)

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 2, January 2017

if*related***then**

relatedSites=retrieveUnvisitedSite(*page*)Out put *relatedSites*

close

close

close

The above algorithm is intended for In-Depth Searching process. Using this algorithm In-Depth Searching is implemented, the input for performing this process is Seed site/Base site for which we perform the In-Depth Searching. This process gets started when the no. of links to be extracted is below a certain threshold value. Then it will take the base site as a input & perform extraction of links. Then it will check unvisited sites & start exploring the link by extracting the hyperlinks present inside the link. The internal links extracted are fed to the Queue to reprise this process. Perform the same for each & every link present in the Queue. Then after performing this classify the links weather they are related or not. If they are related fed the relevant sites to the site database.

C. Cumulative Site Prioritize Algorithm:

input *:siteFrontier*

output: *searchable forms and out-of-site links*

HQ =SiteFrontier.CreateQueue(HighPriority)

LQ=SiteFrontier.CreateQueue(LowPriority)

while*siteFrontier is not empty* **do**

if*HQ is empty***then**

HQ.addAll(LQ)

LQ.clear()

close

site= HQ.poll()

related= organizeSite(site)

if*related* **then**

doSiteAnalysis(site)

Output forms &OutOfSiteLinks

siteRanker.rank(OutOfSiteLinks)

if*forms is not empty***then**

HQ.add (OutOfSiteLinks)

Close

else

LQ.add(OutOfSiteLinks)

Close

Close

Close

The above algorithm is intended for Cumulative site prioritize. This algorithm helps in applying the ranking mechanism for the site. The list of links to perform deep crawling iskept in the site frontier in the form of a Queue. The site frontier is given as input to this algorithm & it provides whether the given site has searchable forms or not & also the Out of site links from the base site/seed site given from the site frontier. For implementation of this algorithm we require two queues. The queues implemented in this framework are priority queues. HQ&LQ are the names of queues. Create HQ&LQ with respect to the site frontier.

D. Site Analysis:

Site Analysis is the 2nd stage of Keen Crawler, in which deep exploration on a particular site is made.This site analysis phase practices a form focused crawler to efficiently gather the deep web inside a provided web link. These crawlers' focuses on an exact topic specified & hunt for the searchable-forms inside a deep link. If any forms exist it catches the hyperlinks involved to that form & starts travelling into the link by mining the links. The mined links are provided as yieldin the form of matchedurls only when they are related which will be taken care by the algorithms. [7] presented a short overview on widely used microwave and RF applications and the denomination of frequency bands. The chapter start outs with an illustrative case on wave propagation which will introduce fundamental aspects of high frequency technology.

E. Crawling Strategies

Stop-early:

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 2, January 2017

Prior work [8] witnessed that 72% interfaces and 94% web databases are found inside the profundity of three. Accordingly, in-site searching is madein breadth-first approach to attainextensivecoverage of web directories. Additionally, in-site searching employs the subsequent stopping criteria to avoid infertile crawling.

*KC1:*The extreme depth of crawling is touched.

*KC2:*The extreme crawling pages in each depth are reached.

KC3: A predefined number of forms found for eachdepth are reached.

*KC4:*If the crawler reaches a predefined amountof pages without searchable forms in one penetration, it drives to the next penetrationstraight.

*KC5:*The crawler takes a predefined amountof pages in total without searchable forms. KC1 limits the extreme crawling depth. For each phase this paper, set some stop criteria (*KC2, KC3, and KC4*). A global one (*KC5*) confines the total pages of infertile crawling.

F. Feature Selection:

The Crawling includes penetrating through different categories of pages for keen crawler. To avoid ambiguity in exploring, the individual feature spaces for site and link are built & utilized

The Feature space for a site is built (FSS) is defined below:

$$FSS = \{U_s, I_s, T_s\}$$

Where U_s – Site URL.
I_s– Anchor tags in the site.
T_s– Text around Site URL.

The Feature space for a link is built (FSL) is defined below:

$$FSL = \{R_l, I_l, T_l\}$$

Where R_l – Path corresponding to URL.
I_l– Anchor tags in the link.
T_l– Text around URL of the link.

For selecting features, Term frequency (TF) is calculated for each term & weight (*w*) of the term (*t*) is calculated as follows:

$$w_{d,t} = 1 + \log tf_{d,t}$$

Where $tf_{d,t}$denotes the frequency of term *t* existing in document *d*. Here document *d* can be U, R, I, T related to site or link.

G. *Ranker:*

Site ranker & link ranker ranks the site & links respectively with the help of ranking mechanisms. This paper uses Cosine similarity as measure to rank the most relevant site as well as links. Cosine similarity. To Prioritize the Urls fetched by the keen crawler two features namely SS(Site Similarity)& SF(Site Frequency). Site similarity is obtained by Cosine Similarity & SF is the number of sites that are most repeating.

SS is given as follows:

For Site:-$SS(s) = Sm(U_s, U_{ns}) + Sm(I_s, I_{ns}) + Sm(T_s, T_{ns})$;

For Link: -$SS(l) = Sm(R_l, R_{nl}) + Sm(I_l, I_{nl}) + Sm(T_l, T_{nl})$;

Where Sm – cosine similarity function.
$\{U_{ns}, I_{ns}, T_{ns}\}$ –

U_{ns} – URLof the new site.
I_{ns}– Anchor tags in thenew site.
T_{ns}– Text around URL of the new site.
$\{R_{nl}, I_{nl}, T_{nl}\}$ –
R_{nl} – Path corresponding to new URL.
I_{nl}– Anchor tags in the new link.
T_{nl}– Text around URL of the new link.

Cosine Similarity is given as follows:

$$Sm(V_1.V_2) = \frac{V_1.V_2}{|V_1|*|V_2|}$$

Where V_1 and V_2 are two vectors.

SF is given as follows:

$$SF = \sum_{\text{known list of Sites}} I_m$$

WhereI_m= 1 if siteappeared in known deep web sites,
OtherwiseI_m= 0.

IV. RESULTS

TheResultsobtained for *Keen Crawler*are discussed below. Keen Crawler consists of an option to limit the Urls crawling to the main or base site. This option is utilized when the user wants to crawl only to a particular site and doesn't need to move out of site to fetch the links. This crawler also provided with option like maximum Urls to be crawled for matching.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 2, January 2017

Id	Urlname	Sword	rank	murl	Dt
1	http://in.bookmyshow.com/	Movies	1	50	25/09/2017 13:57:08
2	http://in.bookmyshow.com/	Movie	1	50	25/09/2017 13:57:15
3	http://www.cricbuzz.com/	Score	2	50	25/09/2017 13:57:50
5	http://www.vnrvjiet.ac.in/	placements	3	50	25/09/2017 14:02:13
6	http://www.vnrvjiet.ac.in/	Courses	1	50	25/09/2017 14:11:31
7	http://www.vnrvjiet.ac.in/	admissions	1	50	25/09/2017 14:14:18
8	http://www.vnrvjiet.ac.in/	Exams	1	50	25/09/2017 14:18:09
9	https://www.tutorialspoint.com/	Java	1	50	25/09/2017 18:15:39
10	http://www.tutorialspoint.com/	Java	1	50	25/09/2017 18:15:43

User can maintain the list of crawled Urls that are matching in a log file. This crawler is provided with information like time to crawl each site. The crawler is specified with a search string to start the process on a specific base site.The crawler contains another feature to rank the sites that are crawled.

Table 1. Shows table of crawled site details against the search keyword along with its ranking.This table also contains the maximum urls crawled.These information is given as a dataset to Weka[10] & J48 decision tree is used to obtain a decision tree for visualization.Fig.2. Represents the bar graph representation of crawled sites ranking for the search keyword.

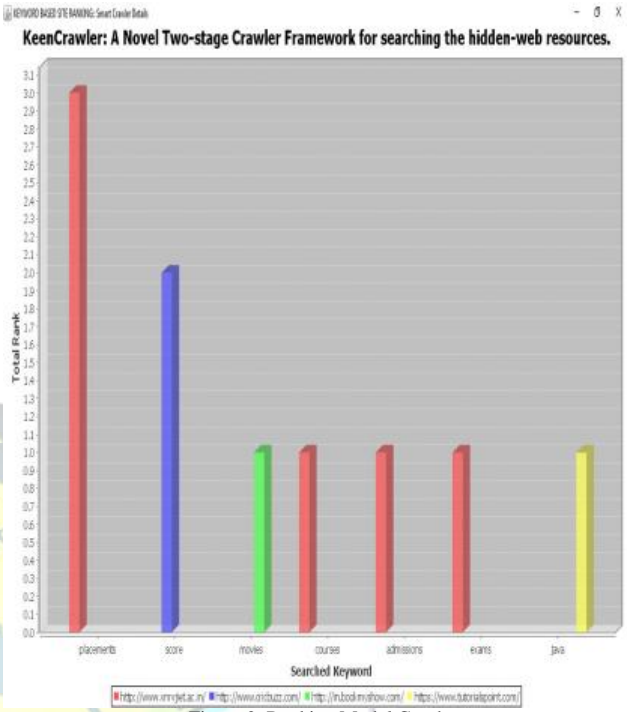

Figure 2. Ranking Model Graph

Table 1. Crawled Site Details

V. CONCLUSION

Keen Crawler is implemented in thispaper, it consists of two Site Discovering phase& Site Analysis phase. In the Site Discovering phase all the urls pertaining to the base site are obtained. In case the base site doesn't have enough links to crawls then In-Depth Search is started with the help of In-Depth Search Algorithm & Cumulative site prioritize Algorithm is used to discover the relevant sites as well as out of site links. In Site Analysis phase the links inside a specific link or site are fetched depending whether the site have searchable forms or not. A FFC is used to find the searchable forms inside a link. The sites are classified based on the ranking given by the site ranker & link ranker. The ranking is given based on the site similarity utilizing cosine similarity for this purpose. Keen crawler provides a wider coverage area for the deep hidden sites as the search starts from centre pages.

REFERENCES:

[1] Feng Zhao, Chang Nie, Jingyu Zhou and H.Jin, "Smart Crawler a two stage Crawler for efficiently harvesting Deep Web interfaces",Vol. 9, No. 4, July/August 2016.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 2, January 2017

[2] Michael K. Bergman. White paper: The deep web: Surfacinghidden value. *Journal of electronic publishing*, 7(1), 2001.

[3] Kevin Chen-Chuan Chang, Bin He, and Zhen Zhang. Towardlarge scale integration: Building a metaquerier over databases on the web. In *CIDR*, pages 44–55, 2005.

[4] Denis Shestakov. Databases on the web: national web domainsurvey. In *Proceedings of the 15th Symposium on InternationalDatabase Engineering & Applications*, pages 179–184. ACM, 2011.

[5] Luciano Barbosa and Juliana Freire.Searching for hidden-webdatabases. In *WebDB*, pages 1–6, 2005.

[6] Luciano Barbosa and Juliana Freire.An adaptive crawlerfor locating hidden-web entry points. In *Proceedings of the16th international conference on World Wide Web*, pages 441–450.ACM, 2007.

[7] Christo Ananth, [Account ID: AORZMT9EL3DL0],"A Detailed Analysis Of Two Port RF Networks - Circuit Representation [RF & Microwave Engineering Book 1]", Kindle Edition, USA, ASIN: B06XQY4MVL, ISBN: 978-15-208-752-1-7, Volume 8, March 2017, pp:1-38.

[8] JayantMadhavan, David Ko, ŁucjaKot, VigneshGanapathy,Alex Rasmussen, and Alon Halevy. Google's deep web crawl.*Proceedings of the VLDB Endowment*, 1(2):1241–1252, 2008.

[9] SoumenChakrabarti, KunalPunera, and MallelaSubramanyam.Accelerated focused crawling through online relevancefeedback. In *Proceedings of the 11th international conferenceon World Wide Web*, pages 148–159, 2002.

[10] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer,Peter Reutemann, and Ian H. Witten. The weka data miningsoftware: an update. *SIGKDD Explorations Newsletter*, 11(1):10–18, November 2009.