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

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

Efficient Web Searching Using Crawler

Monika Bhide[1], Sunita Kerure[2], Amruta Patil[3,] Prof. M.A.Shaikh[5]

Student, Dept. of IT, Rajarshi Shahu College of Engineering, Tathawade, Savitribai Phule Pune University, Pune, India[1234]

Professor, Dept. of IT, Rajarshi Shahu College of Engineering, Tathawade, Savitribai Phule Pune University, Pune, India[5]

ABSTRACT: The advancement of significant web is speedy as appear differently in relation to surface web, so there has been extended excitement for frameworks which can profitably isolate significant web interfaces. In any case, in light of the huge measure of benefits in web with their dynamic nature, removing capable site pages through significant web grow multifaceted nature. We propose a structure with two-stage, to be particular SmartCrawler, which capably look for significant web interfaces. In the primary stage, SmartCrawler refuse passing by considerable number of pages, arrange incredibly pertinent webpage pages to get accurate results and performs site based examining for website pages with the help of web records. In the second stage, SmartCrawler finishes snappy in-site unearthing so as to look for most noteworthy associations with an adaptable association situating. To discard slant on setting off to some significantly essential associations in covered web registries, we arrange an association tree data structure to achieve more broad extension for a website. Our test results on a course of action of specialist's spaces exhibit the preparation and accuracy of our proposed crawler framework, which successfully recuperates significant web interfaces from considerable scale regions and fulfills higher harvest rates than various crawlers.

KEYWORDS- Smart Crawler, Deep web, two-stage crawler, Reverse Searching, Site Locating, ranking, adaptive learning, Site Prioritizing.

I. INTRODUCTION

A Web Crawler is furthermore termed as a robot or a 8-legged creature is a system for the mass downloading of site pages. Web crawlers are used for different purposes. Most unmistakably, they are one of the crucial sections of web crawlers, systems that gather a corpus of pages and document them which allow customers to issue questions against the rundown and find the site pages that match the request. A related use is web chronicling (an organization gave by e.g., the Internet record [3]), where tremendous courses of action of site pages are every so often assembled and archived for youngsters. A third use is web data mining, where site pages are explored for authentic properties, or where data examination is performed on them (a case would be Attributor [5], an association that screens the web for copyright and trademark infringements). Finally, clients can submit standing inquiries to web checking organizations, or triggers, and they relentlessly crawl the web and tell clients of pages that match those request. The significant (or disguised) web implies the substance lie behind searchable web interfaces that are not requested by means of web lists. Considering extrapolations from a study done at University of California, Berkeley, web contains about 91,850 terabytes of significant web and around 167 terabytes of surface web in 2003 [1]. Later studies assessed that 1.9 zettabytes were come to and 0.3 zettabytes were eaten up worldwide in 2007 [2], [3]. An IDC report assesses that the total of all automated data made, rehashed, and used will accomplish 6 zettabytes in 2014 [4]. An imperative portion of this colossal measure of data is assessed to be secured as sorted out or social data in web databases — significant web makes 96% of all the substance on the Internet, which is 500-550 times greater than the surface web [4], [3]. This data contain a vast measure of critical information and components, for instance, Infomine [5], Clusty [3], and BooksInPrint [4] may be possessed with building a record of the significant web sources in a given space. Since these components can't get to the selective web arrangements of web crawlers (e.g., Google and Baidu).

DOI: 10.15680/IJIRCCE.2016. 0404298

II. Literature Survey

At the point when information is looked, a huge number of results show up. Clients don't have the ingenuity and stretch to experience every single page recorded. So web search tools have a difficult task of sorting out the outcomes, in the request of enthusiasm to the client inside of the principal page of appearance and a snappy rundown of the data gave on a page [1]. Recovering powerful substance from the Web is a pivotal errand since it vigorously impacts the apparent adequacy of a web index. Clients regularly take a gander at just a couple top hits, making the accuracy accomplished by the positioning calculation of foremost significance. Early web indexes positioned pages essentially in light of their lexical similitude to the question. The key system was to devise the best weighting calculation to speak to Web pages and question in a vector space, so that closeness in such a space would be connected with semantic pertinence [3]. Web Crawler is a system/programming or robotized script which peruses the World Wide Web in an orderly, mechanized way [4]. Crawlers have bots that bring new and as of late changed sites, and after that list them. By this procedure billions of sites are crept and recorded utilizing calculations (which are normally all around protected mysteries) contingent upon various Factors. A few business web indexes change the variables regularly to enhance the web crawler's process [1]. The fundamental strategy executed by any web creeping calculation takes a rundown of seed URLs as its information and over and over executes the accompanying steps [3].

1. Remove a URL from the URL list.
2. Download the corresponding page.
3. Check the Relevancy of the page.
4. Extract any links contained in it.
5. Add these links back to the URL list.
6. After all URLs are processed, return the most relevant page.

A* algorithm combines the features of uniform-cost search and pure heuristic search to efficiently compute optimal solutions. A* algorithm is the Best First Search algorithm in which the cost associated with a node is $f(n) = g(n) + h(n)$, where $g(n)$ is the cost of the path from the initial state to node n and $h(n)$ is the heuristic estimate of the cost of the path from node n to the goal node. Thus, $f(n)$ estimates the lowest total cost of any solution path going through node n. At each point a node with lowest f value is chosen for expansion. Ties among nodes of equal f value should be broken in favor of nodes with lower h values. The algorithm terminates when a goal node is chosen for expansion [5]. Adaptive A* uses A* Search to find shortest paths repeatedly. It uses its experience with earlier searches in the sequence to speed up the current A* Search and run faster than Repeated Forward A* [6]. In a given state space with positive action costs, the task of Adaptive A* is to repeatedly find cost-minimal paths to a given set of goal states. The searches can differ in their start states. Also, the action costs of an arbitrary number of actions can increase between searches by arbitrary amounts.

Adaptive A* uses informed h-values to focus its searches. The initial h-values are provided by the user and must be consistent with the initial action costs. Adaptive A* updates its h-values after each search to make them more informed and focus its searches even better [7].

III. SMART CRAWLER IDEA

A web crawler or creepy crawly is a PC program that scans the WWW in sequencing and computerized way. A crawler which is in some cases alluded to insect, bot or specialists is programming whose reason it is performed web slithering. The essential design of web crawler is given beneath (Figure1). More than 13% of the activity to a site is produced by web look [1]. Today the span of the web is a large number of a large number of pages that is too high and the development rate of website pages are likewise too high i.e. expanding exponentially because of this the primary issue for web index is arrangement this measure of the extent of the web. Because of this substantial size of web affects low scope and web search tool indexing not cover 33% of the openly accessible web [12]. By examining different log documents of various site they found that most extreme web solicitation is produced by web crawler and it is on a normal half [15]. Creeping the web is not a programming undertaking, but rather a calculation outline and framework plan challenge in light of the web substance is substantial. At present, just Google cases to have recorded more than 3 billion site pages. The web has multiplied each 9-12 months and the changing rate is high [1, 2, 3].About 40% website pages change week after week [5] when we consider delicately change, however when we consider changing by 33% or more than the changing rate is around 7% week after week [7].

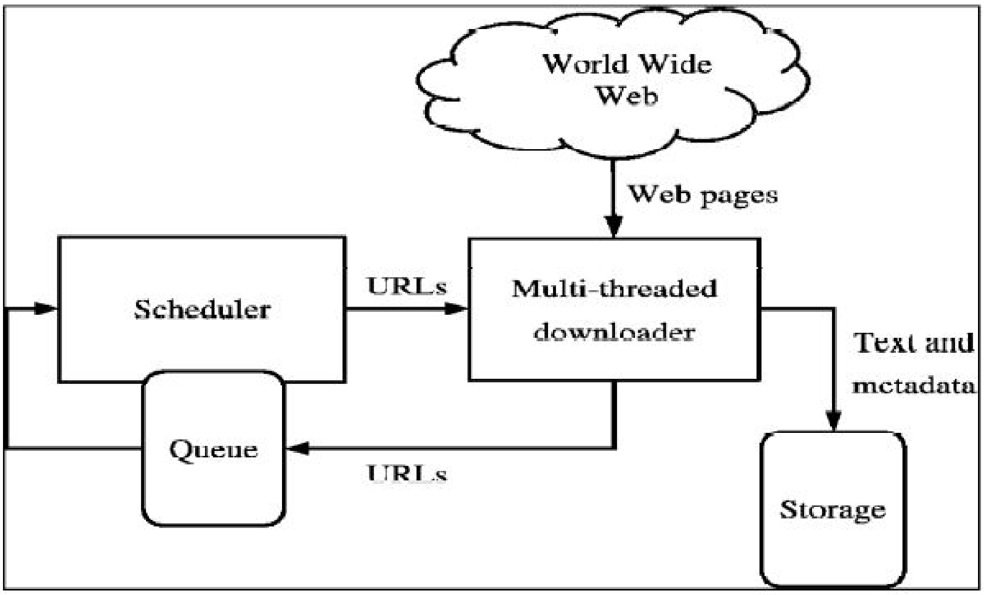


Fig No 01 Smart Crawler Idea

Specialists are growing new booking approach for downloading pages for the internet which ensures that, regardless of the possibility that we need don't download all the pages despite everything we download the most imperative (by the client perspective) ones. As the extent of Internet information develops, it will be exceptionally indispensable to download the critical ones to begin with, as it will be difficult to download every one of them. Whatever is left of the paper is composed as takes after. Section2 is clarified about basics of web creeping. Section3 gives insight about web crawler systems with outline. Section4 has basic examination with tables. Research degree is in area 5. Conclusion and references are finally.

WEB CRAWLER STRATEGIES

1 Breadth First Search Algorithm

Broadness first calculation chip away at a level by level, i.e. calculation begins at the root URL and quests the every one of the neighbors URL at the same level. In the event that the fancied URL is discovered, then the hunt ends. On the off chance that it is not, then pursuit continues down to the following level and rehashes the procedures until the objective is come to. At the point when all the URLs are checked, however the goal is not discovered, then the disappointment reported is created. Broadness first Search calculation is for the most part utilized where the target lies as a part of the depthless parts in a more profound tree. [6][13].

2 Depth First Crawling Algorithms

Profundity first pursuit calculation is a more valuable hunt which begins at the root URL and navigates profundity through the youngster URL. To start with, we move to one side most kid if one or more than one kid exist and cross profound until no more is accessible (Figure3). Here backtracking is utilized to the following unvisited hub and procedures are reimbursed in comparable way [9]. By the utilization of these calculations creators ensures that every one of the edges, i.e. all URL is gone to once breathe [10]. It is extremely effective for inquiry issues, yet when the youngster is huge then this calculation goes into a vast circle [8].

3 Page Rank Algorithm

By Page rank calculation web crawler decides the significance of the website pages in any site by the aggregate number of back connections or references in giving page [10]. The page rank of a gave website page is computed as Relatedness between the networks pages are considered by the Page Rank calculation. The website page whose number of info connection is high is considered of more significance in respect to other site page, i.e. interest level of the page to another. At the point when the quantity of information connection is expanded, then intrigue level of a page clearly additionally increments. Accordingly, the aggregate weighted entirety of information connections characterizes the page rank of a page [11]

4 Online Page Importance Calculation Algorithms

On-line Page Importance Computation (OPIC) in this method, to find that importance of any page in web site, i.e. each page has a unique cash value that is equally distributed to all output links, initially all pages in any website have the same cash and it is equal to 1/n. The crawler will start downloading web pages with higher cashes in each and every stage and cash will be distributed among all the pages it points when a web page is downloaded. Unfortunately, by the use of in this method, each web page will be downloaded many times so that the web crawling time also increase [14]

5 Crawling the large sites first

In 2005 Ricardo BaezaYates et al "Slithering a Country: Better Strategies than Breadth First for Web Page Ordering" perform tests in approx 100 million site pages and find that creeping the expansive webpage first plan has for all intents and purposes most valuable then on-line page significance calculation. The web crawler fined as a matter of first importance un –crawled website pages to discover high need site page for picking a site, and begins with the destinations with the vast number of pending pages [3].

6 Crawling through URL Ordering

Junghoo Cho et al "Effective Crawling Through URL Ordering" find that a crawler is to choose URLs and to check from the line of referred to URLs to discover more imperative pages first when it visits prior URLs that have stay content which is like the driving question or connection separation is likewise short to a page and that kind of website pages to be known vital [20].

7 Batch-page rank

Clump page rank In This technique first computation of an estimation of Page rank by the assistance of pages seen in this way, every N page downloaded. After those again next N pages are chosen to download are those website pages with the most elevated assessed Page rank [20].

8 Partial-page ranks

Halfway page rank This is likewise like bunch page rank, however lack of concern between Page rank recalculations of N pages in cluster page rank and fractional page rank; a transitory page rank is doled out to every single new page by utilizing the aggregate entirety of the Page rank of the site pages indicating it and it is partitioned by the aggregate number of out-connections of those pages [21].

9 By HTTP Get Request and Dynamic Web Page

It is a Query based Approach to minimize the Web Crawler or insect Traffic by utilizing HTTP Get Request furthermore Dynamic Web Page. As indicated by the creator it is an inquiry based way to deal with illuminate all reports on the site by web crawler utilizing by Dynamic website page furthermore HTTP GET Request [17]. What's more, crawler download just upgraded site pages after the last visit.

10 Using Customized Sitemap

Expanding the Efficiency of Crawler by the Using Customized Sitemap. At the point when a crawler returning to the sites and find what site pages have been upgraded or recently included since last visit, then there is no compelling reason to download the complete site inevitably. With this plan, it will be less tedious for web crawlers to keep up the freshness of downloaded sites utilized via web indexes. [18].

11 By the use of filter

"A Novel Web Crawler Algorithm on Query based Approach with Increases Efficiency" The creators proposed an alter approach for creeping by the utilization of a channel and this is a question based methodology. Channel dependably diverts the overhauled website pages and crawler downloads all upgraded site pages after LAST_VISIT [19].

IV. SYSTEM ARCHITECTURE

To gainfully and feasibly find significant web data sources, SmartCrawler is made with two stage outline, site finding and in-site page exploring, as showed up in Figure. The essential site discovering stage finds the most appropriate site for a given subject, and after that the second in-site exploring stage uncovers searchable structures from the site.

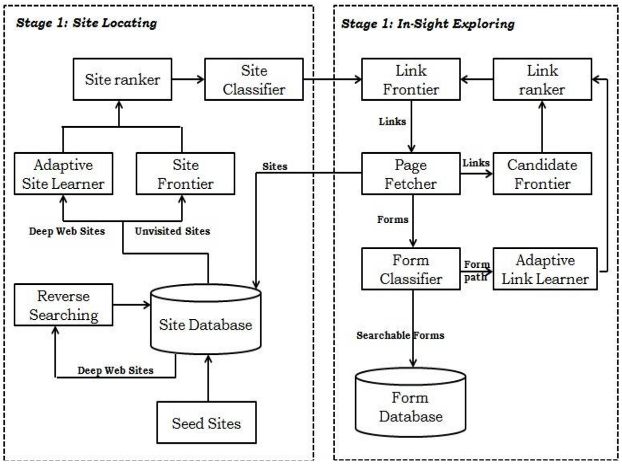


Fig. 2System Architecture

Specifically, the site discovering stage starts with a seed set of destinations in a site database. Seeds destinations are candidate districts given for SmartCrawler to start crawling, which begins by taking after URLs from picked seed destinations to examine distinctive pages and diverse territories. Right when the amount of unvisited URLs in the database is not precisely an edge in the midst of the crawling process, SmartCrawler performs "reverse looking for" of known significant locales for center pages (exceedingly situated pages that have various associations with various zones) and feeds these pages back to the site database.

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

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

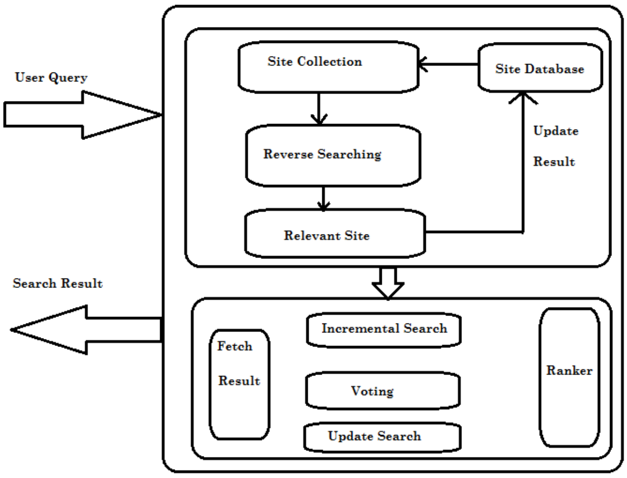


Fig. 3 System Architecture for Incremental Search.

RANKING SEARCH RESULT STEPS

1 Start
2. Initialize rank for all webpage to 0.
3. When User clicks on any of the URL its IP Address, Date of visit, time and No. of clicks are stored in the database and a particular rank is given to the website.
4. If user clicks on the same website again within 24 hours of time than algorithm check again the IP address, time, Date and no of clicks which are already stored in the database if the IP address of user match than it increases only no. of clicks in the database and do not increase ranking of the website.
5. If user clicks on different website, all process of ranking algorithm repeat and a different rank is given to the website.
6. The process repeated and according to its rank position of websites is given in the search result list.
7. Stop.

V. ALGORITHMS & TECHNIQUES USED

Algorithm 1: Reverse searching for more sites.
Input : seed sites and harvested deep websites
Output : relevant sites

```
while # of candidate sites less than a threshold do
// pick a deep website
site = getDeepWebSite(siteDatabase,seedSites)
resultP age = reverseSearch(site)
links = extractLinks(resultP age)
foreachlink in links do
page = downloadPage(link)
relevant = classify(page)
if relevant then
relevantSites=extractUnvisitedSite(page)
Output relevantSites
end
end
```

```
end
```

Algorithm 2: Incremental Site Prioritizing.

Input : siteFrontier

Output : searchable forms and out-of-site links

```
HQueue=SiteFrontier.CreateQueue(HighPriority)
LQueue=SiteFrontier.CreateQueue(LowPriority)
while siteFrontier is not empty do
    if HQueue is empty then
        HQueue.addAll(LQueue)
        LQueue.clear()
    end
    site = HQueue.poll()
    relevant = classifySite(site)
    if relevant then
        performInSiteExploring(site)
        Output forms and OutOfSiteLinks
        siteRanker.rank(OutOfSiteLinks)
        if forms is not empty then
            HQueue.add (OutOfSiteLinks)
        end
        else
            LQueue.add(OutOfSiteLinks)
        end
    end
end
```

A web crawler (in like manner termed as a robot or a bug) is a structure, an undertaking that crosses the web with the final objective of mass downloading of site pages in an electronic manner. Web crawlers are prominently one of the standard fragments of web crawlers that accumulate a corpus of pages or makes a copy of all the went to pages, record them, and license customers to issue request against the rundown, give speedy chases and find the site pages that match the inquiries. Speaking with countless servers and name servers, inching is considered as the most fragile application since it is outside the capacity to control of the system. Crawler makes after particularly clear strides yet to a great degree convincing work in bolster, checking of the downloaded joins moreover the endorsement of HTML codes as tails It starts with the once-over of URL's to visit, called seeds and downloads the site page.

VI. WORKING

A. MODULE INFORMATION

1. Two-stage crawler.

It is trying to find the profound web databases, since they are not enlisted with any web crawlers, are generally inadequately appropriated, and keep continually evolving. To address this issue, past work has proposed two sorts of crawlers, non-specific crawlers and centered crawlers. Non specific crawlers bring every searchable shape and can't concentrate on a particular point. Centered crawlers, for example, Form-Focused Crawler (FFC) and Adaptive Crawler for Hidden-web Entries (ACHE) can naturally seek online databases on a particular theme. FFC is outlined with connection, page, and shape classifiers for centered slithering of web structures, and is stretched out by ACHE with extra segments for structure sifting and versatile connection learner. The connection classifiers in these crawlers assume a urgent part in accomplishing higher creeping proficiency than the best-first crawler However, these connection classifiers are utilized to anticipate the separation to the page containing searchable structures, which is hard to gauge, particularly for the postponed advantage joins (interfaces in the long run lead to pages with structures). Subsequently, the crawler can be wastefully prompted pages without focused structures.

2. Site Ranker

At the point when joined with above stop-early arrangement. We take care of this issue by organizing very pertinent connections with connection positioning. Notwithstanding, connect positioning might present inclination for exceptionally pertinent connections in specific registries. Our answer is to assemble a connection tree for an adjusted connection organizing. Figure 2 delineates a sample of a connection tree developed from the landing page of http://www.abebooks.com. Interior hubs of the tree speak to catalog ways. In this illustration, servlet registry is for element demand; books index is for showing distinctive inventories of books; Amdocs catalog is for demonstrating help data. By and large every index as a rule speaks to one kind of records on web servers and it is worthwhile to visit joins in various registries. For connections that just contrast in the inquiry string part, we consider them as the same URL. Since connections are regularly dispersed unevenly in server registries, organizing joins by the significance can conceivably predisposition toward a few indexes. For example, the connections under books may be allocated a high need, since "book" is an imperative element word in the URL. Together with the way that most connections show up in the books catalog, it is very conceivable that connections in different registries won't be picked because of low significance score. Thus, the crawler might miss searchable structures in those registries.

3. Adaptive learning

Versatile learning calculation that performs online component choice and utilizations these elements to naturally build join rankers. In the site finding stage, high pertinent locales are organized and the creeping is centered on atopic utilizing the substance of the root page of destinations, accomplishing more exact results. Amid the in site investigating stage, applicable connections are organized for quick in-site seeking. We have performed a broad execution assessment of Smart Crawler over genuine web information in 1representativedomains and contrasted and ACHE and website based crawler. Our assessment demonstrates that our slithering structure is extremely compelling, accomplishing considerably higher harvest rates than the best in class ACHE crawler. The outcomes additionally demonstrate the adequacy of the converse looking and versatile learning.

B. INPUT AND OUT PUT

Parameter	Methods of searching	Output	Result Accuracy
Document Corpus/Title of document	Document Based	Searched pages	80%
Audio file or speech	Audio based	Search result	40%
Image content	Image based	Parsed result	30%
Location name and expected search	Location based searching	Location wise search result	50%
Concept(Topic ,Text)	Text / Concept based	Relevant search from query	90%

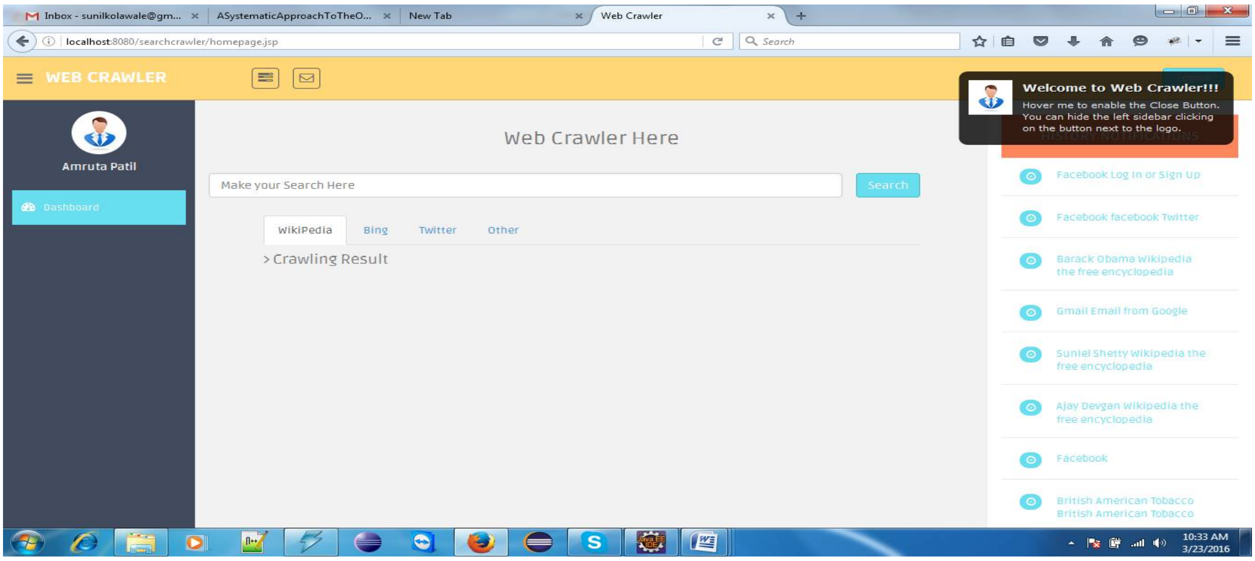
Search with Proposed Method:

We are using concept based searching method from above which is uses text as input query. Text comes useful in all cases of search like audio by converting audio to text, location in textual and reading document content for finding expected search from internet.

Parameter	search	Result	Accuracy
URL	By matching URL text against query	URL containing user search	90%
Title of URL	Comparing title with user search input	Relevant title for search	90%
Description of URL (Words, sentences)	Reading content of document from search result	Matching content	99%

Example:

Query input	result	accuracy
facebook	Facebook(80%),face(10%),book(10%)	95%
Cloud computing	Cloud Computing (70%), cloud (20%), and technology (10%)	90%
Pune	Pune(70%),relevant information(30)	85%



Welcome to Web Crawler!!!
Hover me to enable the Close Button. You can hide the left sidebar clicking on the button next to the logo.

Web Crawler Here

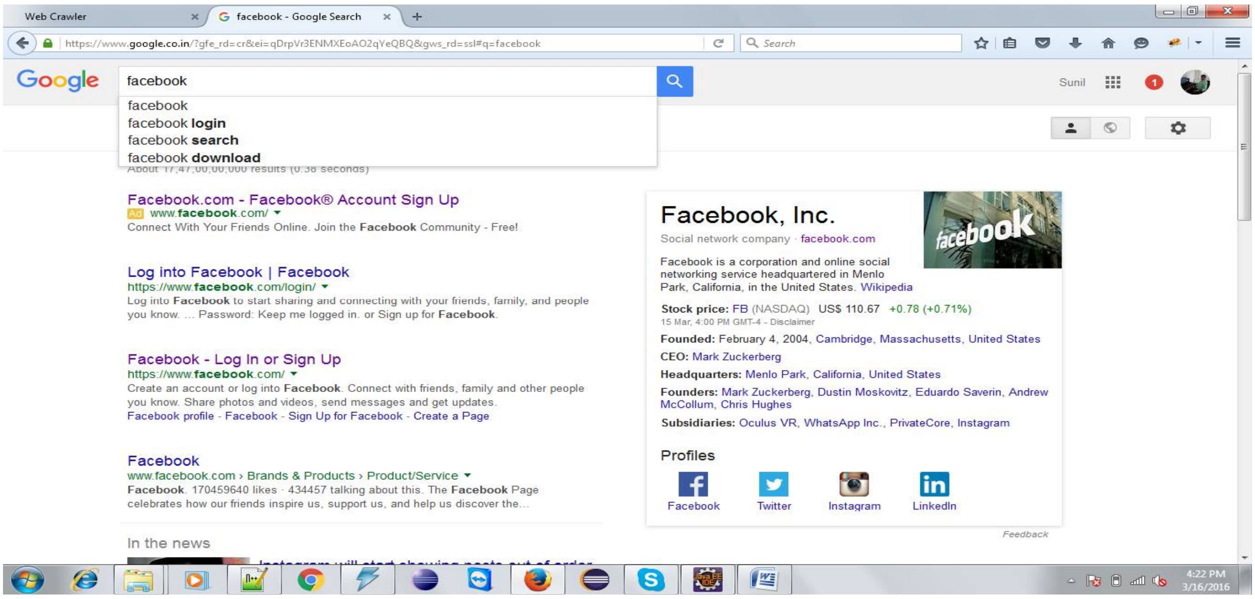
Make your Search Here

> Crawling Result

HISTORY NOTIFICATIONS

- Facebook Log In or Sign Up
- Facebook facebook Twitter
- Barack Obama Wikipedia the free encyclopedia
- Gmail Email from Google
- Suniel Shetty Wikipedia the free encyclopedia
- Ajay Devgan Wikipedia the free encyclopedia
- Facebook
- British American Tobacco British American Tobacco

Our Browser:



facebook

facebook
facebook **login**
facebook **search**
facebook **download**

About 17,47,00,00,000 results (0.38 seconds)

Facebook.com - Facebook® Account Sign Up
Ad www.**facebook**.com/
Connect With Your Friends Online. Join the **Facebook** Community - Free!

Log into Facebook | Facebook
https://www.**facebook**.com/login/
Log into **Facebook** to start sharing and connecting with your friends, family, and people you know. ... Password: Keep me logged in. or Sign up for **Facebook**.

Facebook - Log In or Sign Up
https://www.**facebook**.com/
Create an account or log into **Facebook**. Connect with friends, family and other people you know. Share photos and videos, send messages and get updates.
Facebook profile - Facebook - Sign Up for Facebook - Create a Page

Facebook
www.facebook.com › Brands & Products › Product/Service
Facebook. 170459640 likes · 434457 talking about this. The **Facebook** Page celebrates how our friends inspire us, support us, and help us discover the...

Facebook, Inc.
Social network company · facebook.com

Facebook is a corporation and online social networking service headquartered in Menlo Park, California, in the United States. Wikipedia

Stock price: FB (NASDAQ) US$ 110.67 +0.78 (+0.71%)
15 Mar, 4:00 PM GMT-4 - Disclaimer

Founded: February 4, 2004, Cambridge, Massachusetts, United States

CEO: Mark Zuckerberg

Headquarters: Menlo Park, California, United States

Founders: Mark Zuckerberg, Dustin Moskovitz, Eduardo Saverin, Andrew McCollum, Chris Hughes

Subsidiaries: Oculus VR, WhatsApp Inc., PrivateCore, Instagram

VII. EXPERIMENTAL SETUP AND RESULT

In this paper, we propose a convincing gathering framework for significant web interfaces, to be particular SmartCrawler. Sharp Crawler is a connected with crawler including two stages: successful site finding and balanced in-site exploring. Proposed reverse searching computation used for finding significant site pages from interest, which deliver result until result matches watchful customer look for. Proposed count find most huge mission from web for customer request. It results connected with and unfocused customer look. This is more relevant request without outside site outline change. Proposed web record produce customer required chase data. We are using taking after computation for bringing required customer look.

Graph-

It demonstrates that the system load brought on by M1 is somewhat higher than the one created by the customary crawler S1 on the off chance that we don't permit page pressure. This is because of the overhead of crawler relocation. On the off chance that we permit M1 to pack the pages before transmitting them back, M1 beats S1 by a component of 4. The remaining bars in Figure 20 demonstrate the outcomes for portable crawlers M2 to M4. These crawlers use remote page determination to lessen the quantity of pages to be transmitted over the system in light of the doled out watchword set. In this way, M2, M3, and M4 mimic subject particular Web slithering as required by subject particular web crawlers.

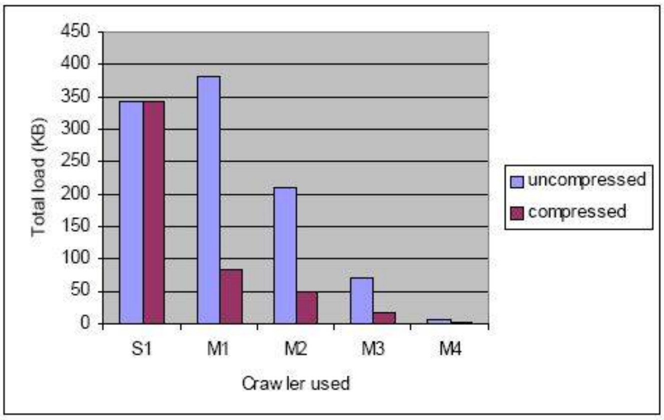


Figure 4: Benefits of Remote Page Selection.

Benefits of Remote Page Filtering

To gauge the real advantages of remote page sifting we adjusted our crawler calculation such that just a specific rate of the recovered page substance is transmitted over the system. By changing the rate of page information saved by the crawler, we can reenact distinctive classes of utilizations. Figure 21 condenses our estimations for a static arrangement of 50 HTML pages. Every bar in Figure 21 shows the system load brought about by our portable crawler M1 relying upon the channel degree doled out to the crawler. The system burden is measured with respect to the system heap of our conventional crawler S1.

DOI: 10.15680/IJIRCCE.2016. 0404298

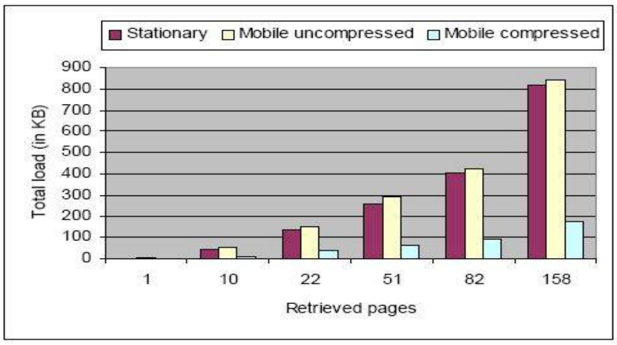


Figure 5: Benefits of Remote Page Filtering.

VIII. CONCLUSION

Web inching is getting more hugeness consistently. So uprooting Deep web ending up being more crucial which is proposed in this paper. The study did in light of wet blanket asking for reveals that the incremental crawler performs better and is all the more successful in light of the way that it licenses re-appearance of pages at changed rates. Inching at other environment, for instance, disseminated has been a future issue to be overseen.

REFERENCES

[1] Peter Lyman and Hal R. Varian. How much information? 2003. Technical report, UC Berkeley, 2003.

[2] Roger E. Bohn and James E. Short. How much information? 2009 report on american consumers. Technical report, University of California, San Diego, 2009.

[3] Martin Hilbert. How much information is there in the"information society"? Significance, 2012.

[4] Idc worldwide predictions 2014: Battles for dominance – and survival on the 3rd platform. http://www.idc.com/research/Predictions14/index.jsp, 2014.

[5] Michael K. Bergman. White paper: The deep web: Surfacing hidden value. Journal of electronic publishing, 2001.

[6] Yeye He, Dong Xin, VenkateshGanti, SriramRajaraman, and Nirav Shah. Crawling deep web entity pages. In Proceedings of the sixth ACM international conference on Web search and data mining, 2013.

[7] Infomine. UC Riverside library. http://lib-www.ucr.edu/,2014.

[8] Clusty's searchable database dirctory. http://www.clusty.com/, 2009.

[9] M. Bergman, "The Deep Web: Surfacing Hidden Value,"J. Electronic Publishing, vol. 7, no. 1, pp. 7-11, 2001.

[10] K.C.-C. Chang, B. He, C. Li, M. Patel, and Z. Zhang, "StructuredDatabases on the Web: Observations and Implications," SIGMOD Record, vol. 33, no. 3, pp. 61-70, 2004.

[11] A. Wright, "Searching the Deep Web," Comm. ACM, vol. 51, no. 10, pp. 14-15, 2008.

[12] J. Madhavan, L. Afanasiev, L. Antova, and A. Halevy, "Harnessing the Deep Web: Present and Future," Proc. Fourth Biennial Conf. Innovative Data Systems Research (CIDR), 2009.

[13] J. Madhavan, D. Ko, L. Kot, V. Ganapathy, A. Rasmussen, and A. Halevy, "Google's Deep Web Crawl," Proc. VLDB Endowment, vol. 1, no. 2, pp. 1241-1252, 2008.

[14] Mauricio C. Moraes, Carlos A. Heuser, Viviane P. Moreira, and Denilson Barbosa, "Prequery Discovery of Domain-Specific Query Forms: A Survey" vol. 25, no. 8, august 2013.

[15] Hai Dong, Member, IEEE, and Farookh Khadeer Hussain "Self-Adaptive Semantic Focused Crawler for Mining Services Information Discovery" VOL. 10, NO. 2, MAY 2014