International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 03 Issue: 12 | Dec -2019 ISSN: 2582-3930

Web Crawler Architecture for Collection of Structured and Unstructured Data

Author: kalpesh r. rakholia, Research Scholar,Saurashtra University, Rajkot
Correspondent Author: Jayesh N. Zalavadia , Research Guide, Saurashtra University,Rajkot

ABSTRACT:

Recently, services furnished to buyers are more and more being combined with big data such as less expensive shopping, personalized advertisement, and product recommendation. With the growing significance of big data, the internet crawler that collects records from the internet has also emerge as important. However, there are two troubles with present net crawlers. First, if the URL is hidden from the link, it can now not be accessed by way of the URL. The 2d is the inefficiency of fetching greater data than the person wants. Therefore, in this paper, via the Casper.js which can manage the DOM in the headless browser, DOM tournament is generated through having access to the URL to the hidden link. We also endorse an sensible net crawler system that permits customers to make steps to fine-tune each Structured and unstructured facts to carry solely the facts they want. Finally, we exhibit the superiority of the proposed crawler device through the overall performance assessment results of the present internet crawler and the proposed internet crawler.

Introduction

Low price shopping, customized optical using the latest big data various fields based on big data such as high and product recommendation And the importance of big data It's getting bigger. This causes the data to the importance of collecting web crawlers is also high. Hyperlinks make web sites look like spider webs Text, images, and videos in a connected web environment Collect and embed data contained in Web documents Links to other web sites and back An automated program that collects is called a web crawler . Access hyperlinks from other web sites here The first problem is derived . HTML The hyperlink inside is an A tag . A tag href in Has a last name and this property It has a URL address . Crawl web through this address Russia can access web sites . Problem java's You can use the crypt function to hide the URL address. Concealed because the URL for existing web crawlers for the cannot access and collect. The second problem is that Crawler can access structured data such as tags and text

Separate unstructured data such as video and audio There is a limit in the house . In this paper, Casper.js based web crawler system Suggest . The proposed web crawler system is hidden In the Headless Browser environment to access the URL Casper.js can be used to access hidden URLs Extensions Pro are features you can grab and browser Via grams, the type of data you want (

text , non Video , audio, etc.) , set the specific area you want to capture and You can do it . Collection based on the set value Have . This paper is organized as follows . Existing Web in Chapter 2 Explore the crawler's features, issues, and related technologies . Web crawler using Casper.js suggested in Chapter 3 . Suggest a stem . In Chapter 4 , we'll look at the Proposed through performance between unused web crawler systems Your web crawler system in Chapter 5 The conclusions and future research directions are discussed .

Related Research

2.1 General Web Crawler Techniques

A web crawler is an organizational , automated way A computer program that navigates the Web . Web crawler The Web crawl tasks (web crawling) or spa It's called Ethering , and it's a bot or software It is a form of agent . Web crawlers are largely ordinary web

There are crawlers and distributed web crawlers .

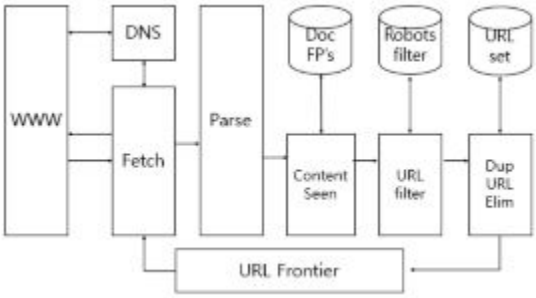

Fig. 1 conventional web crawler architecture

It shows the overall flow of . The default behavior of web crawlers are as follows . The crawler Get a URL from the URL Frontier module and use the http protocol To retrieve a web page at that URL using a call Start with . Then save temporarily in the Fetch module

Save the Web page to the cattle . Text in the Parse Module Extract the link and the text sent to the Indexer The . In the case of links, they should be added to the URL Frontier About whether Content Seen, URL Filter, Duplication Judgment is made through the URL Element modules [1]. But the problem is concealed in the link URL if there In spite of URL being added to URL Frontier

There is a hidden limit that cannot be added .2.2 Distributed Web Crawler Technology Lock all of your Web documents from around the world into the regular web crawler Because rolling is virtually impossible You must have a web crawler . Distributed Web crawler is greatly 2 a Divided into two, one of which Google used Centralized (Centralized) method and the other is P2P (or Fully-Distributed) used in or elsewhere a method [2,3,4,5].

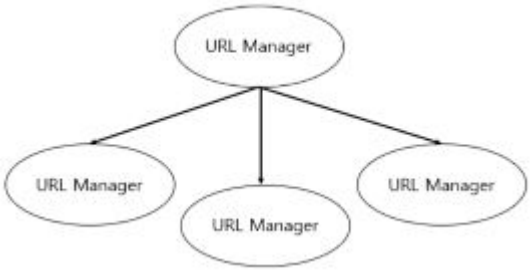

Fig. 2 distributed web crawler structure

this is the centralized dispersion Represents a web crawler structure . Centralized, distributed web crawler URL Manager is Acts like a server, and the crawler It is a structure to make . Download the document from the crawler Extract Out Link URLs and pass them to URL Manager

Give URL Manager is of the downloaded document URL is Check URLs to eliminate URL duplication . Normal web

Crawler URL duplication and URL part of the administration URL Manager does it for you . P2P method each as Crawler a completely independent structure Have . P2P (peer to peer, or fully distributed) methods 'S crawler behaves like a regular web crawler . each Crawler downloads the document and retrieves the OutLink URL .

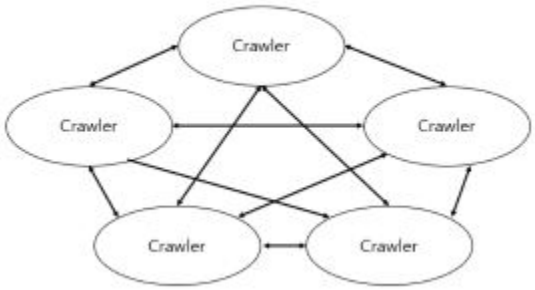

Fig. 3 P2P distributed web crawler structured diagram

Extract and URL de-duplication, all from each crawler Works independently . To do this, Of Crawler downloaded to managed URL list Must be mutually exclusive . Otherwise each other crawler phenomenon to download the same document in will occur . How to solve this each Crawler is mutually exclusive of URL Domain to download Manage by dividing by . That is , it is free do- main management only, and the rest belong to the URL is different Pass it to Crawler . So that each Crawler can work independently [6]. 2.3 Web content extraction technology Web content extraction technology analyzes information from web documents The title , author , publication date and body Provides the ability to extract with copper . Web content extraction The system automatically generates rules to extract content Automatic content extraction rules with devices that extract only content Rule Generator to generate , given web Navigation to remove navigation content from a document Sean Content Eliminator (Navigation Content Eliminator), Content extraction rules Content by keyword similarity comparison Content extractor (Core Context Extractor) is composed of [7].

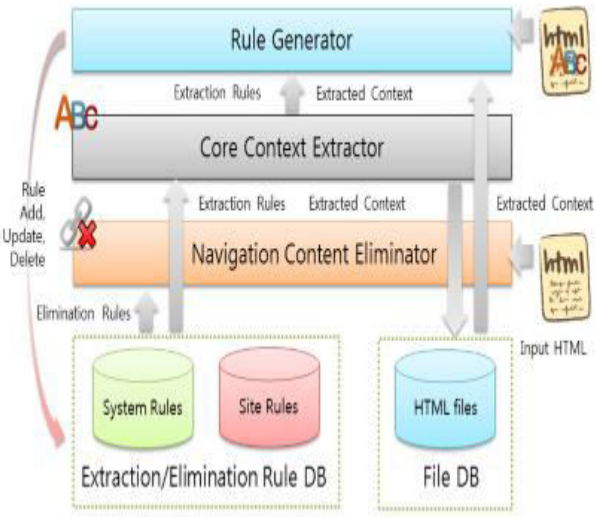

Fig. 4 system composition diagram of web contents extractor

It shows the system configuration . Existing researched web crawler technology Web crawls for unstructured and unstructured data collection Different systems can collect data from hidden URLs . Existing web crawlers and in order to Headless Browser for The difference is that they are organized together . And this Collect

multimedia data for efficient operation Define roles to collect URLs and hidden URLs .

3. Web crawlers for collecting structured and unstructured data system

3.1 Web crawlers for importing structured and unstructured data System configuration

The proposed web crawler can retrieve structured and unstructured data. efficiently collect features [8] and conceal from links Crawling system that can also collect web pages of URLs System . The server is based on Node.js running asynchronously. The existing URL to crawl through the Crawler Service and Casper Service to crawl hidden URLs Include . The client wants the user to crawl Sets the specific area and type of data you want to collect. Extensions and crawled results Includes monitoring . Figure Fig. 5 is the proposed web size Shows the configuration of the roller system , Table 1 is to offer Detailed description of the Web crawler system . Proposed web crawler web server Node.js is single High performance supporting non-blocking IO based on thread Server . But because it is a single thread , There is a limit to crawling web pages . this To solve the problem, run Cluster were utilized . Cluster is a Node.js module that shares a server's port. Can create and process multiple processes . Through this Cluster Crawler Service Worker in writing By creating a red and crawling it, Solved . The Collection Rules module of the Web Server Service Crawl rules and URLs received from clients, or Gets a seed, a list of XPaths . XPath (XML

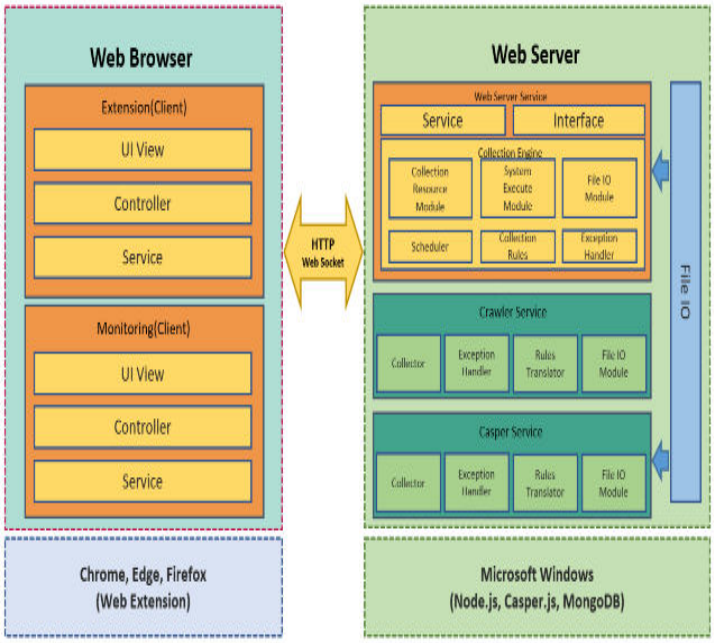

Fig. 5 web crawler system configuration diagram

Path Language) is a W3C standard extension language. Use the syntax specified above the path through the document's structure To describe how to place and process items Language . Crawl rules contain information that the crawler will crawl. Type is defined . The crawler collects according to the data type defined in the rule Determine the tag of a DOM . HTML tags are multi- Because it is described differently according to the deer data, If the video tag is an image, if is an img tag In the case of music, we divide in audio tag and describe in tag Downloaded to the server through the multimedia URL Manage through the Collection Resource Module .

3.2 Web crawler crawl rule definition schedule management function

The Scheduler module builds on the Collection Rules module. Takes a rolling rule and a seed as arguments and the value of the seed is a URL If it passes to Crawler Service Worker and crawls Perform the ring . XPath if the Web Server Service Of File IO host to the module URL, XPath, crawl rules Save it as a JSON file . When the save is complete System Execute Casper Service with Execute module . Casper Service is saved as File IO module immediately after execution . Read the JSON file into the Rules Translator module Crawl web pages by interpreting JSON files . Crawler Service Worker or Casper Service are greater Unstructured data collected during rolling is sent to each service. store it on the server via the included File IO module . Crawler The service worker collects when the crawl is complete. List of structured and unstructured data by HTML and rules Create a separate Cluster send calls methods

Web Send the collected result to the Server Service . Casper For per Service , when the crawl is complete, the Crawler HTML and stereotypes, which are the result values collected just like a Service And list of unstructured data via the File IO module And Callback Function is executed by System Execute After notifying the module, the Scheduler module returns a File IO module. Read the value . Scheduler module is one of the web page Every time the crawl is complete, Collection Resource Run the module to create a path for stored unstructured data. After completing the configuration, notify the Schedule module . Schedule Module HTML, structured data , unstructured data Path It values Interface to send to the module Interface module DB Save to . The Scheduler module collects the results collected by each web crawler. Re-extract the link from the HTML of the fruit URL reporting patterns URL whether the classification and URL If not , extract the link's XPath value and reverse If the right URL to extract the value Collection Rules conventional Updates the seed of and schedules its own crawl Make . Under Fig. 6 is the seed of Collection Rules Sequence diagram when the value is a URL . 7 Shows a sequence diagram when the seed value is XpathIndicates

3.3 Data Storage Function Considering Redundant Data

Web crawler crawls in a tangled web environment like a spider web

Table 1: component description table

Division	Component name			Description
Web Server	Web Server Service	Service		A module that handles client requests and responses by HTTP or Websocket.
		Interface		Save data collected by each crawler to DB
		Collection Engine	Collection Resource Module	Module for creating path of unstructured data collected by crawler
			System Execute Module	Window command prompt connection module to run Casper Service
			File IO Module	Module for writing the Collection Rules file to the Casper Service or reading the crawling result data
			Scheduler	Based on URL seed defined in Collection Rules Crawler Service Walker schedule management and Casper service schedule management module
			Collection Rules	A module that manages the collection rules received from the Service and passes the rules to the Scheduler
			Exception Handler	Modules that manage exceptions that occur in Web Server Service modules
	Crawler Service	Collector		Conduct rules and URL-based web crawls interpreted by Rules Translator
		Exception Handler		Modules that manage exceptions that occur in Web Server Service modules
		Rules Translator		A module that interprets the Collection Rules received from the Scheduler so that they can be applied to the Collector.
		File IO Module		A module that stores unstructured data collected by the Collector on the Web server.
	Casper Service	Collector		Conduct rules and XPath-based web crawls interpreted by Rules Translator
		Exception Handler		Modules that manage exceptions that occur in Web Server Service modules
		Rules Translator		A module that interprets the Collection Rules received from the Scheduler so that they can be applied to the Collector.
		File IO Module		A module that reads the rules saved by the Scheduler or stores collected data that have been completed by the Collector
Client	Extension	UI View		Screen for defining collection rules and specifying specific areas
		Controller		Logic module for handling events that occur in the UI View
		Service		Module for communicating with Web Server Service
	Monitoring	UI View		Screen for the administrator to check the results that the crawler has collected
		Controller		Logic module for handling events that occur in the UI View
		Service		Module for communicating with Web Server Service

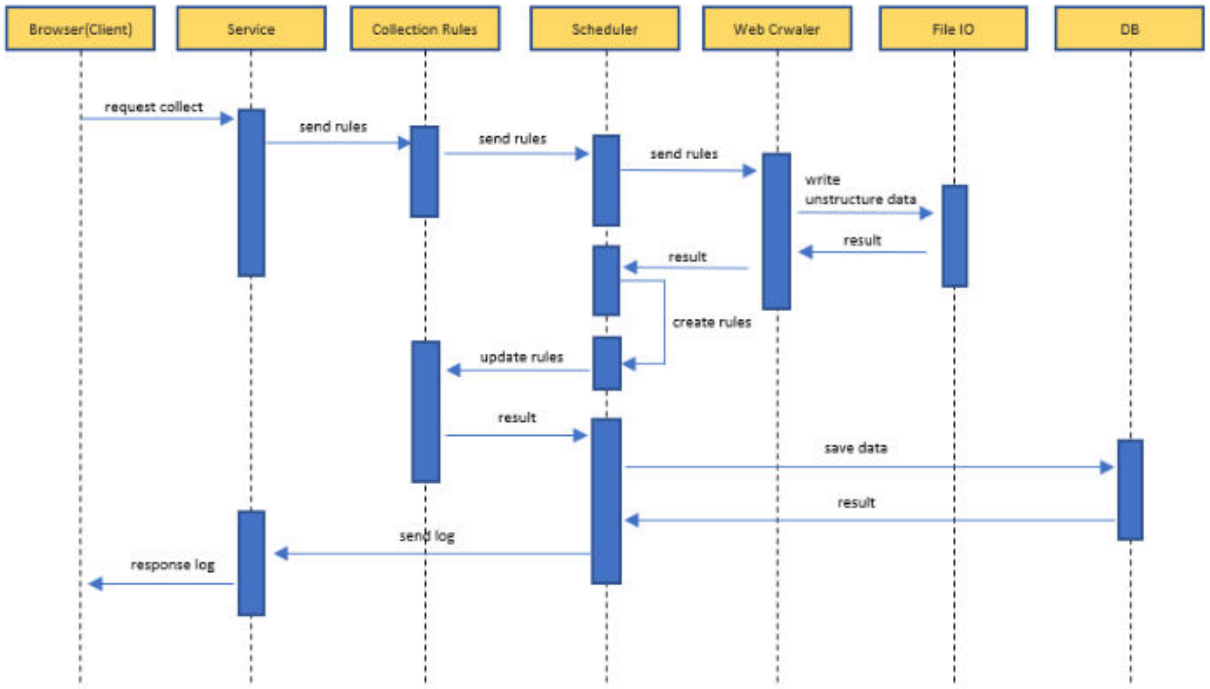

Fig. 6. URL-based web crawler sequence.

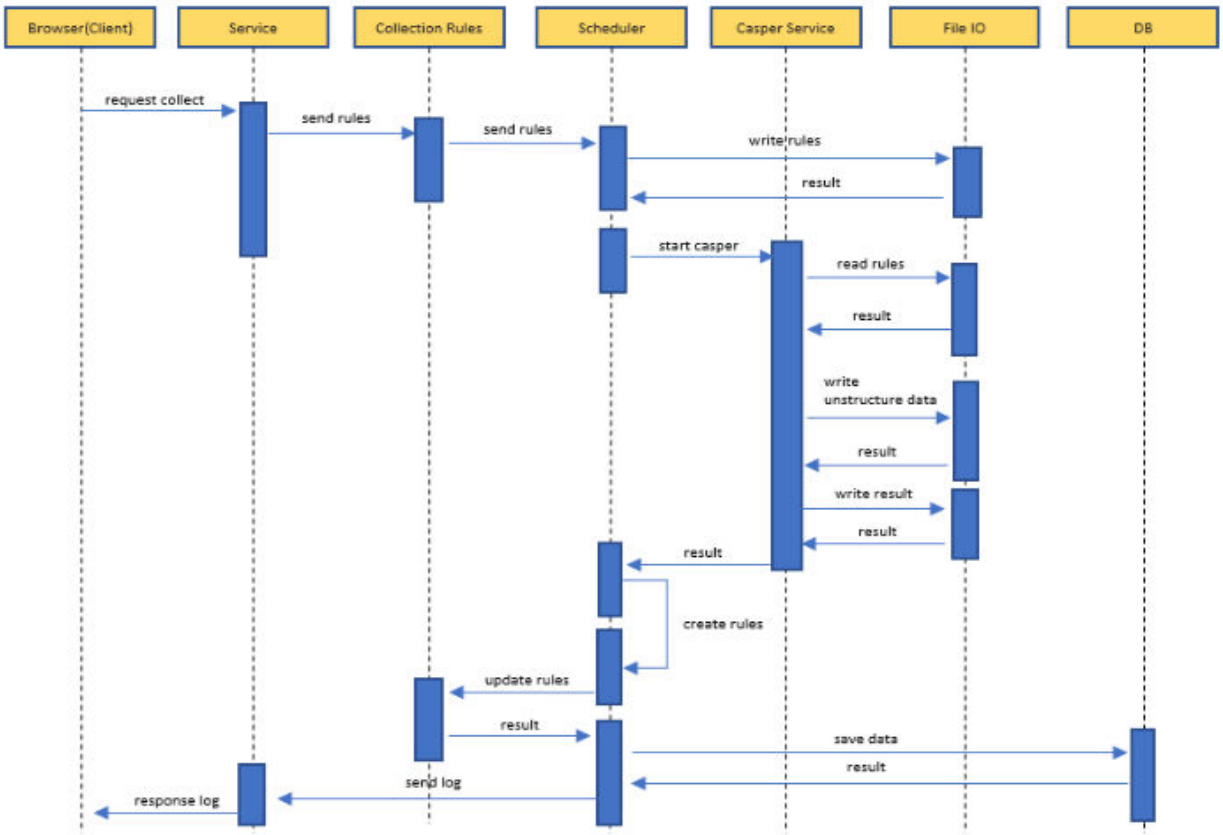

Fig. 7. XPath-based web crawler sequences.

Fig 8 database dataset

Because you can re-crawl the web pages that you Duplicate data may occur . To prevent this The system presented in this paper is based on URL Manage duplicate data with . DB is a crawler Many Read & Write achieved is the Schema less one Properties that can store any type of data Because the features RDB non- Mongo DB to Used . Fig. 8 is stored in DB based on the URL . Indicates the form . Types stored in DB are Host URL Collection and It is divided into Subject URL Collection . “Http: //” statement Host and Subject based on the "/" character after the string The separator and after "/" is Subject URL should be treated as . All Document has _id attribute are automatically granted It has _id that crawlers to navigate through URL is If you are looking for a URL that exists in the DB , the Document Update the Contents property of ment . Host URL Document in the Subject URL Document or Subject URL Document in the Subject URL Document to When browsing, Fig. _Bottom, _top properties of Child Document or parent via _id attribute within Go to the Document . Fig. 9 during the crawl process Flowchart showing the process of checking and saving Picture .

3.4 Crawl Hidden URLs with Casper.js

Concealed URL link consisting of the URL Java Instead It is implemented by calling a script function . That In order to run a function, Attempt access . Above the browser environment Headless because we need to implement a working crawler Browser was used . Headless Browser is a GUI without a CLI (Command Line Interface) A web browser that works . Web pro Gram test automation , screenshots , javascript Library automated testing , data scraping, and more Used

for the purpose . Casper.js is a Headless Browser A framework that provides navigation scripting for . Fig. 10 is a diagram showing the sequence of actions in Casper.js

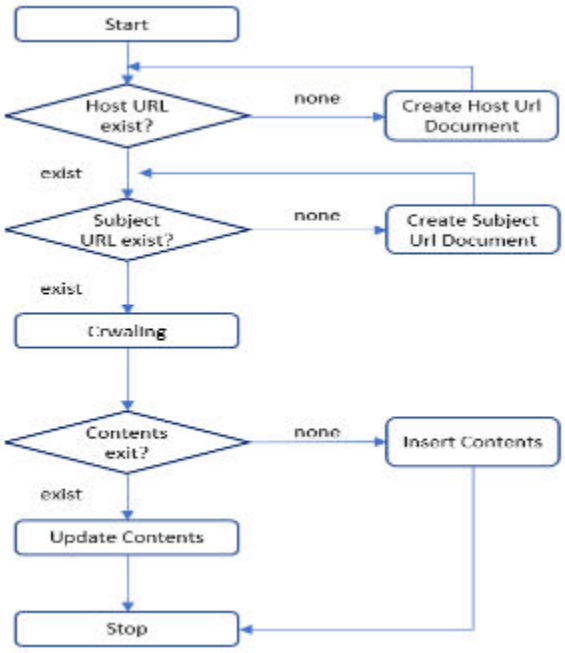

Fig 9: duplicate data validation crawling update flow chart

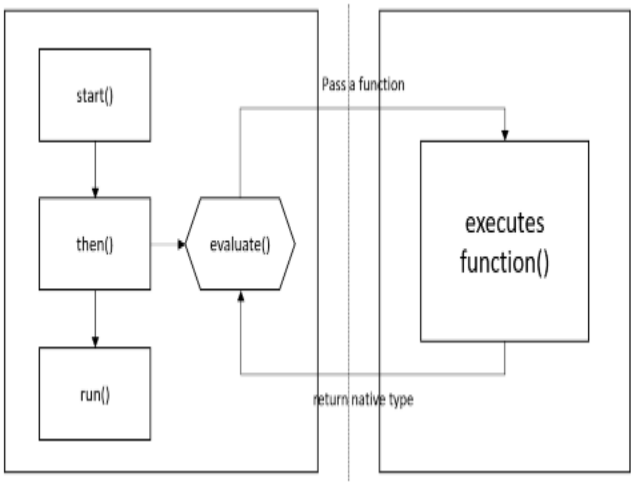

Fig 10: casper.js operating sequence diagram

Casper.js will start by calling the methods Headless Web defined in Collection Rules by running Browser Access the page . And then callback in the then method Perform a synchronous crawler by passing a function . You can call the evaluate method inside the callback function. The . Functions can be defined in the arguments of the evaluate method In this function, the headless browser is connected to the web. You can define DOM events for your page . so Detect link tag with XPath defined in Collection Rules To fire the click event and hide the link You can connect to the URL . Collection when connected Structured or unspecified depending on the data type defined in the Rules Collect type data and links .

3.5 Crawl Collection Area Extraction feature

Crawl rules defining and gathering area extraction technology Implemented through the wser extension . Browser Extension changes the behavior of existing features of the Web browser. Or add a whole new feature for web browsers Program . Extensions are available for Chrome, Firefox, Edge, Works in the Opera browser . Fig. 11 is a suggestion Crawler for collecting structured and unstructured data Shows the extension UI of the system . Fig. 11 is a menu for setting crawling rules. It is . Use as an interface to check checkboxes You can choose the type of data you want to crawl. The . Under the check box, you can set the specific area. There is Sean . If you do not select a specific area Crawling all but select

specific areas No. UI Appears and press a button on the top of the UI The extraction function can be turned ON / OFF . When the function is executed Mouse pointer to the Mouse Up event of the Document object Gets the element of the DOM pointed to by the site . is the back- of the DOM pointed to by the mouse pointer.

ground-color modify the properties to which the user DOM It is implemented to recognize whether the . Mouse Clicking on imported DOM storage elements and No. UI on Convert DOM elements to XPaths and display them in a list . Finally, click the bottom crawl button of XPath host URL of selected data type and specific region Create a crawl rule with the data and pass it to the web server . The rules created through the extension are shown in Fig. 12 It is described in Json syntax as follows: data_type, specif- The ic_area and host_url attributes exist . data_type Has The above property has a Boolean value for each type . Default value is set to false . The specific_area attribute is added by the user It is converted to the XPath syntax of the exported DOM and added .

4. Web crawlers for collecting structured and unstructured data System evaluation

The existing crawler used for performance comparison is (Crawler.js) is Node.js as a module-based web crawler Node.js Is a site where you can install modules available on Is the most popular crawler in . The website for performance comparison is the saurashtra university website. (http://www.saurashtrauniversity.edu.in) and PKM college The homepage (http://www.pkmtc.com) was compared Crawl bulletin posts .

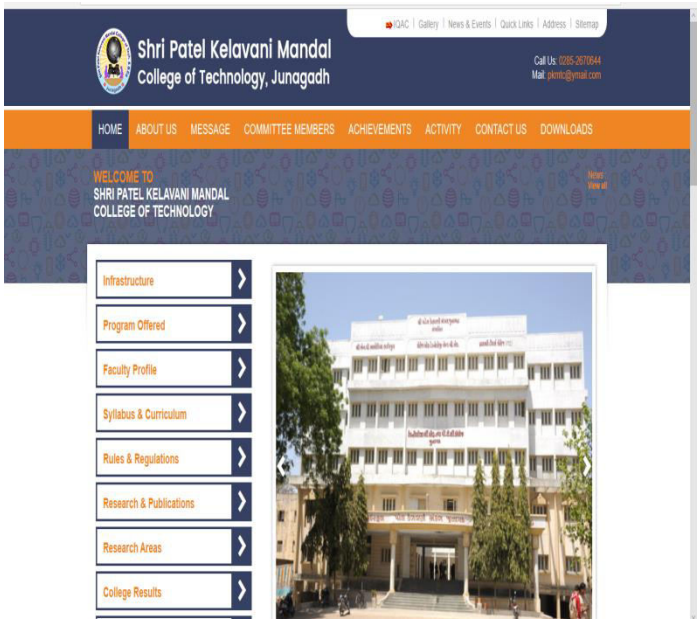

Fig 11: extension UI

A chart showing the number of web pages collected by the crawler The . If you look at the chart, the difference in the number of web pages You can check . This is the

```
{
  "data_type":{
    "text": true,
    "video": true,
    "image": true,
    "audio": false,
    "gif": false
  },
  "specific_area": [
    "//*[@id='PM_ID_themecastBody']/div/div/div/ul"
  ],
  "host_url": "http://www.example.com/main"
}
```

Fig 12: Collection Rules JSON Define

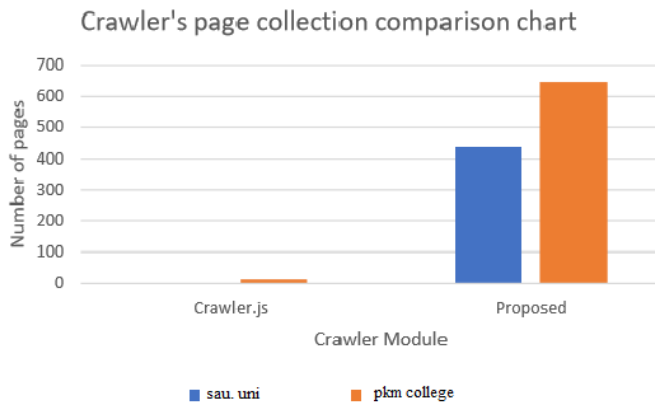

Fig 13: Crawlers page collection comparison chart

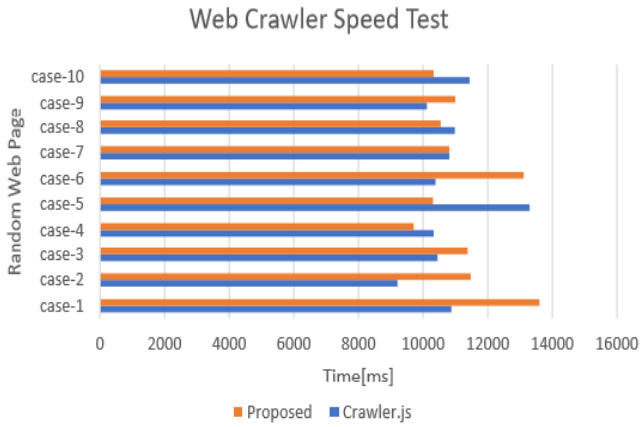

Fig 14: Web Crawler Speed Test

This is because the URL is hidden . Traditional crawlers Consider because hidden URL links cannot be crawled 0 data at the university and 14 data at the college Collected . Rate the speed of the proposed and existing crawlers Was added . Conventional crawlers use the same modules as above. Was used . Such as the web crawler that the paper suggests Since it is Node.js based, I thought it was suitable for comparison . Prior to comparing the test environment 4GM RAM and a 2.20GHz The Intel Core I5, Window 10 servers with the operating system Comparison tests were conducted on the client and the client . Under Fig. 14 lines a total of 20 random web pages

Jung was 10 and proceeded to test one . Processing time is web When to start collecting pages and when to end

The time until was measured . The results are as follows It showed similar performance with the crawler of . But existing 'S crawler fails to fetch links from hidden URLs The speed is similar, but the amount of collection is better. Proved to be a crawler .

5. Conclusion and future research

In this paper, the hidden URL of the existing crawler Problems with inaccessible links, structured data, Problems that can not collect unstructured data separately I've listed and utilized Casper.js to solve it To fire DOM events directly from the Headless Browser The proposed web crawlers to access , text , the structured and unstructured data you want , Image , video , audio data types and features to crawl By selecting specific areas, more precise and efficient The extension function to build a house is presented . In addition, the performance of existing and proposed web crawlers According to the evaluation results, 10 tests with the existing web

crawler In the average case, the average was about 437 ms speed difference .Because this is a value that varies depending on the environment, There is no difference in speed from web crawlers, but In terms of the amount of houses, there is a big difference . You can see . This suggests a crawler Excellence in structured and unstructured data collection capabilities of the system Proved . In the future, the speed will be similar to that of the existing crawler. The performance and the amount of data collected All will improve the excellent web crawler system .

REFERENCE

[1] C.D. Manning, P. Raghavan, and H. Sch¨utze, I ntroduction to I nformation Retrieval, Cambridge University Press, Cambridge, 2008.

[2] Dustin Boswell, Distributed High-Performance Web Crawlers: A Survey of the State Of the Art, 2003.

[3] A. Heydon and M. Najork, “Mercator: A Scalable, Extensible Web Crawler,” World Wide Web, Vol. 2, No. 4, pp. 219-229, 1999.

[4] Allan Heydon and Marc Najork, High-Performance Web Crawling, COMPAQ SRC Reserch Report 173, 2001.

[5] S. Brin and L. Page, “The Anatomy of a Large-Scale Hypertextual Web Search Engine,” Proceeding of the Seventh I nternational World Wide Web Conference, pp. 107-117, 1998.

[6] M.S. Kang and Y.S. Choi, “Design Hadoop Based P2P Distributed Web Crawler,” Proceeding of Korean Society For I nternet I nformation, pp. 199-202, 2010.

[7] D.M. Seo and H.M. Jung, “Intelligent Web Crawler for Supporting Big Data Analysis Services,” J ournal of Korea Contents Association, Vol. 13, No. 12, pp. 575-584, 2013.

[8] Y.H Kim and M.D Chung, “Analysis of Structured and Unstructured Data and Construction of Criminal Profiling System using LSA” J ournal of Korea Multimedia Society, Vol. 20, No. 1, pp. 66-73, 2017.