The Design and Implementation of Web Crawler Distributed News Domain Detection System

I Gusti Lanang Putra Eka Prismana[1,*] Dedy Rahman Prehanto[1]

I Kadek Dwi Nuryana[1]

[1]*Faculty of Engineering, Universitas Negeri Surabaya, Surabaya, East Java 60231, Indonesia*
[*]*Corresponding author. Email: lanangprismana@unesa.ac.id*

ABSTRACT
Spreading data or info through internet to increase the chances of success in a business through analysis of market trends is very common today. Web Crawl is one important thing, so that the incomplete data will not be appeared, and the data received is the most recent data. Exploration Web crawler technology is a technology that downloads web pages via a program. Crawlers and search engines face unpredictable challenges. A focused web crawl is essential for mining the unlimited data available on the internet. The web crawl encountered an undetermined latency issue due to their difference in response time. The proposed research tries to optimize the design and implementation of a distributed news domain detection system on a web crawler. This study proposes a distributed focused crawler because it reduces the appearance of time outs on each website, eliminates backlist capabilities, distributes resources and improves web crawlers work in efficient network bandwidth and storage capacity. The main objective of distributed theory Web Crawler implements crawler scheduling, sorting sites to define URL queues. The crawler is only focused on news data. This research implements URL Gate explorer, which is used as the main bridge of instructions from the database, URL Seed to check all URLs for each news, and get metadata to check each meta data whether there is the same title.

Keywords: *Web crawler, news domain, distributed, focus crawler*

1. INTRODUCTION

Internet is a gathering place for a large amount of information in the world, be it text, media or data in other formats that are usually displayed in a web page. The ease of data accessing is important thing for most business success in the modern world. For companies engaged in marketing, data can be used to determine current market trends, so the most appropriate marketing strategies can be found for each product. Ecommerce-based companies can also use this data for market analysis or simply price comparisons with other ecommerce competitors.

Among the world's total population of 7.5 billion, 3.6 billion are internet users. It means a half of the world's population is on the internet. There are more than 1 billion websites on the World Wide Web (WWW)today. On average, Google processes more than 40,000 search queries per second, which translates to 3.5 billion search queries per day worldwide. Because of information and users on the WWW are growing at a rapid rate, it becomes a challenge for Search Engines to fill all user needs for information searching on interest topics. Typing any query in a Search Engine will generate millions of web documents. Most of these documents are irrelevant to user interests. It is very difficult for users to find relevant information from this huge collection of results. The most searched data on the Internet is news, almost everyone every day needs the latest news. Increasing news data on the internet can be a cost-effective medium of information in Indonesia.

Search Engines use Web Crawlers to collect web pages from the WWW by following hyperlinks on these web pages. Web Crawler is a search engine that works by downloading a web page that passes a hyperlink on that page. Therefore crawlers are often referred to as graphical browsing. The working principle of a crawler is starting with a list of visited URL, then the crawler will visit the URL address one by one until it's finished. Design and implementation of distributed news domain detection system on a web crawler is the first step to obtaining data in the news form which is very complex. Thus, design built can be a data source which easily accessed and used.

In this study, we propose a method for gathering news online. A method called distributed focused crawler. The only result we need is news data. The main components crawlers focused are classifiers, refiners and crawlers. Classification is carried out with the aim of getting a decision on which web page matches the desired problem.

2. BACKGROUND

2.1. Web Crawler

Web crawler does not simply index all the data on the internet, but it will determine which pages need to be crawled, based on other pages that link to those pages and the number of visitors. Although conceptually it can be put into practice easily, in practical theory implementing a crawler is not that simple. Because there are levels of efficiency and other problems. In addition, there are also dangerous crawler applications, for example, crawlers to collect email is used by spammers or collects personal information used in phishing attacks and other labels for data retrieval. In general, crawlers are mostly used to support search engines. In fact, crawlers are the primary consumers of internet bandwidth. They search for search engine pages to create their index. Most famous search engines like Google, Yahoo! and MSN runs a highly efficient and effective universal crawler built to find and collect all pages in the form of all content. Other crawlers, sometimes referred to as special (focused) crawlers, are more targeted. They try to download pages only for certain types or topics. The way a web crawler works involves the following steps:

1. Select a URL from the seed URL set.
2. Enter into the URL Queue.
3. Dequeue URL from Queue
4. Get the web page that matches that URL
5. Extract the new URL from the web page.
6. Enter the newly found URL into the Queue.
7. Extract and store relevant information into a search engine database (index).
8. Continue to step 3 and repeat until the Queue does not empty or exceeds the specified limit.

2.2 Crawler Technique

Web crawlers can explore all the information on the web page itself, and search engines cannot be separated from web crawlers. The main task of the web crawler is to crawl all the data on the Internet, store the necessary information data into local databases and obtain effective information. The theory of realization and the crawler process can be seen from Figure 1

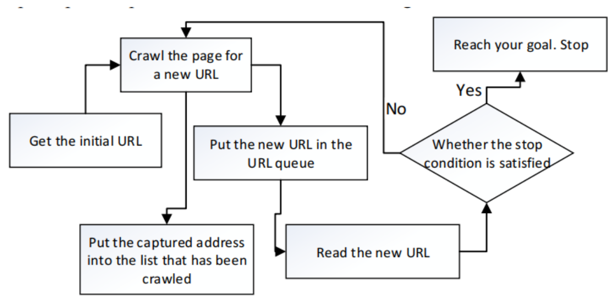
Crawl the page for a new URL
Reach your goal. Stop
Yes
No
Get the initial URL
Put the new URL in the URL queue
Whether the stop condition is satisfied
Put the captured address into the list that has been crawled
Read the new URL

Figure 1 Technique realization process

The crawler mainly includes a downloader, information extractor, scheduler, and crawl queue. The scheduler will enter the URL that has been provided for download, therefore the downloader will get page data from the internet to search and send news extractor results, extractor plan according to the instructions of news extraction to get news and subsequent level in URL. After that, the subsequent level URL waits for the queue to send heavy operation URL, filtering and sorting into lists, after waiting for scheduling calls.

2.3. Characteristics of Crawler

2.3.1. Distributed crawler

Web crawler can be adapted to multiple machines in a distributed area.

2.3.2. Scalability crawler

Due to the large quantity of data, crawling is a slow process. Adding more machines or increasing network bandwidth can improve crawling speed.

2.3.3. Performance and efficiency crawler

The web crawler driving the site for the first time can download all the available files, emphasizing the efficient use of system resources, namely the processor, several machines. Distributed crawlers can be storage and network bandwidth categorized into three parts as follows.

2.3.4. Quality crawler

Get high quality pages that can be used by users. The acquisition of other pages and improve the accuracy of obtaining pages are things web crawlers should give priority.

2.3.5. Freshness crawler

Performs search engine updates, crawls the data independently according to the frequency of changes in each page and database, and crawls new URLs that are done or updated randomly. Like, news, the latest novel.

2.3.6. Extensibility crawler

Designed to be expandable crawlers, with crawlers set up inside a modular architecture. To make adjustments to the new data format and acquisition protocol.

2.4. Type of Web Crawler

2.4.1. Crawler universal

The universal crawler is in charge of downloading the entire content of its Web pages. Also matches what is in the source code of a Web page. Ads or other links can also be downloaded.

2.4.2. Topic crawler (focused web crawler)

The topic crawler only downloads specific topic pages. The difficulty with the topic crawler is how to recognize the page. For example, in the discussion in this thesis, will take news / news. Whereas we all know, the news website of the news provider has "lots of ads", a lot of it is distracting your concentration. Even though what we need is only the core of the news.

2.5. Distributed Web Crawlers

The distributed web crawler will run on multiple computers. Each section will focus on running the crawler. The problem that often occurs in distributed crawlers is how to coordinate and manage activities between nodes, so that each distributed part can run efficiently without causing repetitive work. So, the main key is completing communication coordination between

2.5.1. Master-slave mode

The Master-Slave mode is part of the crawler which works on the principle of utilizing a single host machine and performing check operations of the entire group. Communication with each machine is the responsibility of the host in this method. assigning tasks to each machine, managing a list of URLs to be crawled, and monitoring the working status of each machine to ensure the normal operation of each machine as expected is an important task for the host. Whereas slave machines are only tasked with completing their own tasks and providing reports to the host machine, so that slave machines do not need to communicate with each other.

2.5.2. Autonomous mode

The autonomous mode does not use host to manage the release task. This mode assures the normal execution of distributed crawlers using the interaction between each machine. The communication between each machine has two different forms, which are circular and full unicom communication. Circular communication implies that all machines compose a circular formation that allows one-way communication. While in full unicom communication every machine could communicate with another machines to make the communication network.

2.5.3. Combination mode

Mixed-mode combines the characteristic of master-slave and autonomous mode. In this mode, the host is in charge of task distribution to other machines. However, the slave machines are also capable of interacting with each other and have the task appointment function and the task assignment that failed will be secondary assigned by the host.

3. RESULT AND DISCUSSION

Web crawlers have a basic technique of simulating the browser to design HTTP requests, and the crawler will send requests to a Web server via HTTP. The crawler will analyze and store. Web page, and finish the crawler system crawling job after getting a response from the server. Basically decomposing website pages is a process of reducing the distractions that exist on website pages. On the Internet, all categories of web page information are stored in an HTML framework. In fact, web page denoising is simply the extraction of text from web content. When a theme crawler extracts content on a web page, it needs to decipher the HTML structure of the page in order to effectively extract information from the page. Common methods include parsing the HTML structure of Beautiful Soup and extracting the text data using a regular expression.

3.1. Data Storage Techniques

Basically, data access crawlers have two data storage techniques, namely local file storage techniques and database storage techniques. Some data can be saved directly to local data, but some data can be automatically stored in the database. The database can be configured using the Redis database, which is the key value of a high-performance database. The characteristics of the redis database are that it has an irregular nature and is anti-repetition. Crawlers can generally search for the page content URL that is key in the Redis collection in the database i.e. use the collection for iteration, every time the crawler handles the URL or the page will switch to check if the Redis database already exists, because the Redis database is locked and the value is stored, So speed up this step. will be very high. Second, Redis can save in-memory content files to disk, and each operation is atomic. The advantage of crawlers is that they cannot lose data due to unexpected stops. Figure 2 illustrates the flow of slave states in the proposed solution.

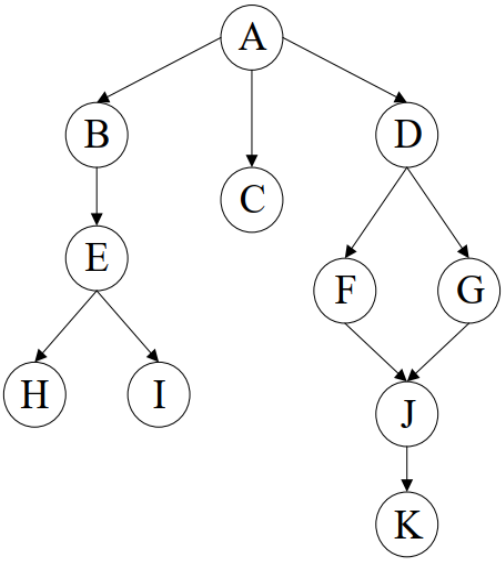

Figure 2 Network link simplified diagram

3.2. Web Search Procedure

In order to design a superior and useful web crawler that can perform its intended distribution tasks across multiple machines simultaneously, it requires a site that distribution can be done independently to share simultaneous access, thus simultaneous distribution will save the capacity of the transmission asset system. Because these crawlers are distributed, the best and most extensive first searches become practical. For the intelligent distributed crawler, we will use the first broader method. The multiple search methods are first utilized in a more global issue. Even in a certain period the value of the website pages can be higher, so that from the beginning of the website pages will continue to be crawling links to the web. The first width of the search procedure makes it possible to get search results that match the tree division level, if the current search is incomplete, it will not move to the next quest. In such a case, the first method of broad search is blind search, which will search all areas of knowledge, reducing efficiency. The first broad search method will be the best choice, if you need focused coverage.

Table 1. A first time crawl route

Number	Path
One	a
Two	b-c-d
Three	e-f-g
Four	h-i-j
Five	k

3.3. Design of Web Crawler Distributed News Domain Detection System

Design is used to produce a web crawler distributed news domain detection system. The dataset used is news data from several sites that will check the URL and meta data. The proposed web crawler architecture is illustrated in Figure 3.

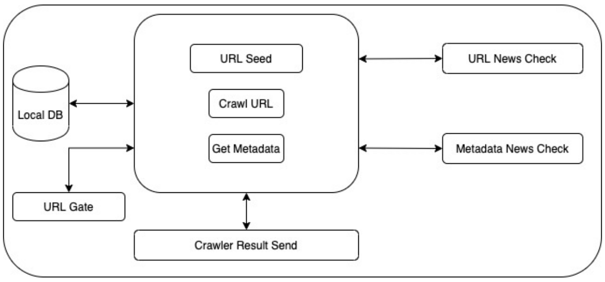

Figure 3 Design of web crawler distributed news domain detection system

The basic principle is URL Gate here becomes the main bridge for data intrusion to be crawled. Next, we will enter the seed URL process which will check the list on all news related href lists and followed by the Get Meta Data process which will check the meta data list whether there are the same news headlines. If there is a list of the same title, the timestamp will be checked. If there is no crowl data, it will be saved directly in the database. The following is the design of the URL Gate web crawler distributed news domain detection system

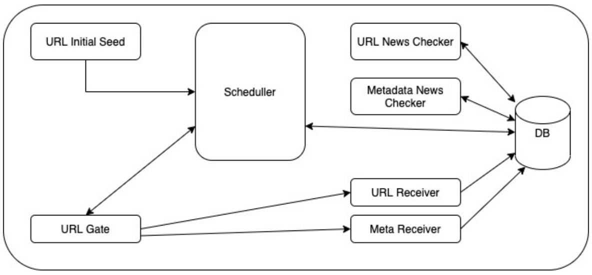

Figure 4 Design URL gate

The scheduler is a simple priority queue based on ordering of the crawl frontier. The crawl frontier is constantly updated by the slaves. The crawl front being a resource accessed by multiple threads is given mutual exclusion using semaphores. A simple First Come First Serve (FCFS) scheduling is used to give access to the scheduler. The priority in the priority queue is the score of the hyperlink being pushed into the queue. The coordinator is the process which maintains and controls the slaves. The coordinator is responsible for configuring the master machine with the available slaves. The coordinator maintains a slave availability system. Being once initialized all available. The coordinator maintains mutual exclusion by monitoring the allotment of the slaves.

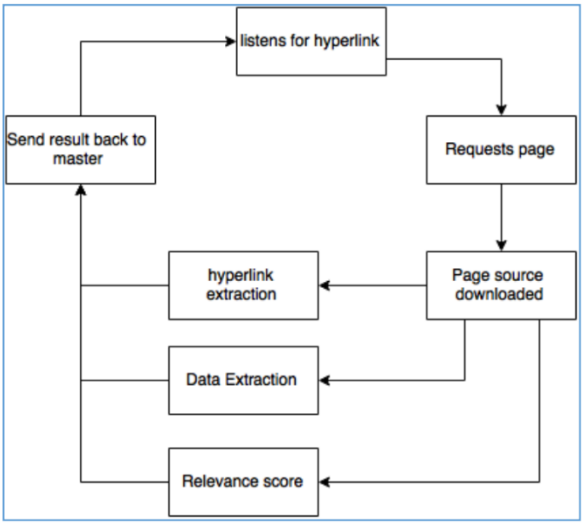

Figure 5 State diagram of slave system of proposed solution

Once the coordinator is requested for a free slave, it responds by giving the requested thread the slave or a busy signal indicating all the slaves are occupied. As the threads complete the processing of a hyperlink using a slave, the thread calls the coordinator to release, which in turn would put the slave back into the available pool. Then slaves are allotted in an FCFS manner, as there is no priority among the several threads that are being processed.

Slave listens scrape requests that are received from the master on a pre-decided port. The master needs to be configured with the set of slaves it works with. On initialization the master reads the configuration file and adds into a queue of available slaves. The slave performs three specific tasks that have varying completion time. The threading and parallelization of these processes over multiple slave systems absorb the variable latency. The tasks of a slave system are to download the source of the requested hyperlink, to extract the content and hyperlinks and to add up the page's relevance score. The relevance score, set of hyperlinks and the content are sent back to the master.

4. CONCLUSION

This research applies a crawler that focuses on distribution to get more news data. This is intended to avoid the kind of time out that exists on every website, eliminate backlist capabilities and be able to distribute slaves' sets are to be resources. In this study, we used several datasets for training, which were taken from several news addresses. Furthermore, the news data crawler will be carried out bridged by the main instructions by the URL Gate. From the data stored in the database, you will check all news href lists by the seed URL, check all news metadata by Get Metadata, which is to detect whether there are the same news headlines, if any, then the timestampt of the news will be detected again, to produce a list of relevant news. This crawl design will implement a simple First Come First Serve (FCFS) Scheduling which is used to grant access to the scheduler.

ACKNOWLEDGMENT

The author would like to thank the support from the Ministry of Education and Culture of the Republic of Indonesia and the State University of Surabaya.

REFERENCES

[1] [1] Aghamohammadi and A. Eydgahi. A novel defense mechanism against web crawlers intrusion. In Electronics, Computer and Computation (ICECCO), 2013International Conference on, pages 269–272. IEEE, 2013.

[2] F. Ahmadi-Abkenari and A. Selamat. An architecture for a focused trend parallel web crawler with the application of clickstream analysis.Information Sciences, 184(1):266–281, 2012

[3] S.SASIREGA, A.Jeyachristy, (2014). "Ontology Based Web Crawler for Mining Services Information Retrieval". International Journal of Computer Science and Mobile Computing, Vol. 3, No. 11, pp.325–330

[4] D. Doran, K. Morillo, and S. S. Gokhale. A comparison of web robot and human requests. In Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pages 1374–1380. ACM, 2013

[5] S.R. Mani Sekhar, et al. 2019. " Optimized Focused Web Crawler with Natural Language Processing Based Relevance Measure in Bioinformatics Web Sources CYBERNETICS AND INFORMATION TECHNOLOGIES, Vol. 19 (2): Page. 146-158

[6] Wan, Y., H. Tong. URL Assignment Algorithm of Crawler in Distributed System Based on Hash.– IEEE International Conference on Networking, Sensing and Control008),2008.Page 1632-1635.

[7] Bal, Sawroop Kaur, Geetha, G. 2016. "Smart distributed web crawler", International Conference On Information Communication And Embedded System(ICICES 2016). Page978-1-5090-2552-7-978-1-5090-2552-7

[8] Yu, Linxuan, et al. 2020. "Summary of web crawler technology research". Journal of Physics: Conference Series. Page 1-7

[9] Zou, Shang-Xuan. 2017. "Distributed Training Large-Scale Deep Architectures". Page 1-10

[10] Butakov, Nikolay, et all. 2016. "Multitenant approach to crawling of online social networks". Procedia Computer Science 101, 2016 , Pages 115 –124.

[11] Allah, Wael A. Gab, et all. 2016. "Performance Analysis of an Ontology Based Crawler Operating in a Distributed Environment". IJSRST. Vol 2. Page 334-339.

[12] S.SASIREGA, A.Jeyachristy, (2014). "Ontology Based Web Crawler for Mining Services Information Retrieval". International Journal of Computer Science and Mobile Computing, Vol. 3, No. 11, pp.325–330.

[13] Xu, Hongsheng, et all. 2018. "Analysis and Research of Distributed Network Crawler based on Cloud Computing Hadoop Platform". Atlantis Press. Advances in Computer Science Research, volume 83. Page 1045-1049.

[14] Maheshwar, Poonam. 2016. "A Cloud-based Web Crawler Architecture". International Journal of Engineering and Management Research. Volume-6. Page 148-152.

[15] Sharma, Gitika, et all. 2016. "EVOLUTION OF WEB CRAWLER ITS CHALLENGES".International Science Press. Vol. 9(11) Page. 5357-5368

[16] YU, JIANKUN, et al. 2016. “A Distributed Web Crawler Model based on Cloud Computing”. Atlantis Press, 2nd Information Technology and Mechatronics Engineering Conference (ITOEC 2016).Hal 276-279.

[17] Wan Shengye, et all. 2017. "Protecting Web Contents Againts Persistent Distributed Crawlers". IEEE ICC 2017 Communication and Information Systems Security Symposium.Page 1-6.

[18] A. Aghamohammadi and A. Eydgahi. A novel defense mechanism against web crawlers intrusion. In Electronics, Computer and Computation (ICECCO), 2013 International Conference on, pages 269–272. IEEE, 2013

[19] Zhao, Feng, Jingyu Zhou, Chang Nie, Heqing Huang, and Hai Jin. "SmartCrawler Two-stage Crawler for Efficiently Harvesting Deep-Web Interfaces." (2015).

[20] A. Amalia, D. Gunawan, A. Najwan and F. Meirina, "Focused crawler for the acquisition of health articles," in 2016 International Conference on Data and Software Engineering (ICoDSE), Denpasar, 2016.

[21] D. Doran, K. Morillo, and S. S. Gokhale. A comparison of web robot and human requests. In Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pages 1374–1380. ACM, 2013.

[22] Plachouras, F. Carpentier, M. Faheem, J. Masanès, T. Risse, P. Senellart, P. Siehndel, and Y. Stavrakas, “ARCOMEM Crawli architecture,” future– inte 541, 2014.