Distributed News Crawler Using Fog Cloud Approach

I. Gusti Lanang Putra Eka Prismana(✉)

Universitas Negeri Surabaya, Surabaya, Indonesia
lanangprismana@unesa.ac.id

Abstract. Technology advanced quickly during the Industrial Revolution. 4.0, makes the internet network also develop rapidly and become larger. So website technology that is constantly changing becomes a big challenge in using large and complex data information on the global Internet. Stand-alone web crawlers have traditionally been difficult to overcome the challenges of rapid information growth, therefore it's challenging to extract a lot of data in a short period. The research will use distributed technology to build a more effective web-distributed news system, to search for news. Crawler systems can work efficiently with Multi-Threads working together, and each node can work efficiently with Multithreading. This study applies a new web crawler fog cloud approach that is considered to be more efficient in navigating URLs by setting according to the domain used and dividing URL limitations into various priority URL queues so that URLs can be dispersed across concurrent crawler operations to get rid of the new building. In particular, the proposed model can effectively utilize resources optimally in the cloud-fog layer by deploying a crawler distribution in the cloud-fog infrastructure to detect news. With the fog cloud, analysis is dynamically distributed across the fog and cloud layers enabling real-time distribution. The research phase of the distributed news crawler starts from URL collection, URL filtering, scheduling, accessing URLs, and extracting news data. This research is focused on developing web crawlers to process distributed news crawlers.

.

Keywords: Web crawler · News · Distributed web crawling · Fog cloud

1 Introduction

Changes in internet technology today are getting faster and faster, this is inseparable from the many availabilities of search engines that make it possible to browse the internet world. There are hundreds or even thousands of search engines on the internet and their capacity always grows from time to time [1]. The explosive development of the Internet makes it difficult for a person to find information that matches what he wants. In related posts, someone often gets information that does not match what is desired. The amount of data on the Internet is becoming larger and the ever-changing technology of websites is a huge challenge to deal with the data dominant on the global Internet.

H. P. Agustin et al. (Eds.): IJCSE 2022, AER 218, pp. 251–260, 2022.
https://doi.org/10.2991/978-94-6463-100-5_26

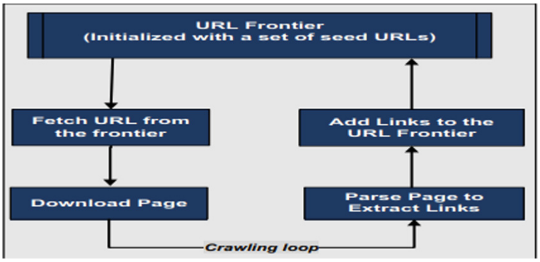
URL Frontier
(Initialized with a set of seed URLs)
Fetch URL from the frontier
Add Links to the URL Frontier
Download Page
Parse Page to Extract Links
Crawling loop

Fig. 1. The basic principle behind Web Crawler.

Simple Web Crawlers are having a hard time dealing with the great challenges of this era of the industrial revolution. In addition, simple web crawlers have not been able to retrieve very large amounts of data in the shortest possible time. As a source of search engine data, web crawlers have an important role in searching for data. Web crawlers have several main indicators that can directly affect the data search process, such as crawl rate, reach, page ranking, index, clock, etc. directly affecting search results.

An important function of a web crawler is to collect the contents of a well-functioning web page which is usually a document in the form of HTML with the part that connects it. We can see the workings of the web crawler in general in Fig. 1.

With this problem, an idea emerged from research to help users in obtaining information according to what they want without having to consume too much time, namely by downloading information from the website automatically and classifying it according to the category of information. One way to download information from a website is to use a web crawler app. Web crawlers are a type of robot or software agent. The main function of the web crawler is to browse the pages of the site and then retrieve those pages for storage. After the information is automatically saved, it will be classified according to the category of news [2].

This study will implement a distributed web crawler using a configurable, efficient, load balancing, and scalability fog cloud approach to determine related posts in each news story, namely by using a crawler distribution. So that the existence of a distributed web crawler can optimize news search and help someone in finding news easily.

Numerous research on web crawlers has been conducted to evaluate the effectiveness of web crawler systems using both simulations and direct implementation methods. Under the heading "News Web Crawler with Xpath Method" in 2015. A program that can record news data was developed because of this research [3]. Additionally, the second web crawler-related study published in 2016 was titled "Smart distributed web crawler." This paper offers a client-server architecture for web crawlers that is based on intelligent distributed web crawlers [4]. The Design and Implementation of a High-efficiency Distributed Web Crawler is the title of the third web crawler-related study published in 2016 in this field. In this study, load-balancing distributed web crawler systems are designed and implemented using distributed technologies [5]. Additionally, the fourth study on web crawlers in 2019 under the working title "Implementation of hybrid P2P networking distributed web crawler utilizing AWS for smart work news large

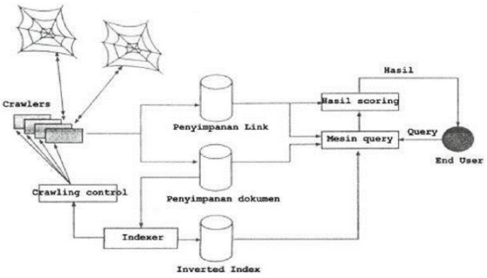

Fig. 2. Search Engine Architecture

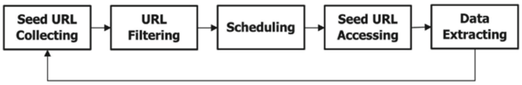

Fig. 3. Web crawling process

data." The paper suggests a hybrid P2P web crawler that can gather web data using an Amazon Web Services (AWS) cloud service platform [6].

2 Related Work

2.1 Search Engine

Search engines (search engines) as shown Fig. 2 are facilities used to explore various data, information, and knowledge that exist on the internet. A search engine is a program that can be accessed via the internet that serves to help computer users in searching for various things they want to know [7]. The American Heritage Dictionary defines a search engine as a software program that searches captures, and displays information from databases.

Searches by Search engines are carried out in a database that stores the text of each page. Text from the page by page is saved into the database server. When performing a search, search engines will search for copies of pages stored in a database containing copies of pages at the time they were last visited. When the link provided is clicked, the address will be given by the search engine server.

2.2 Web Crawler

Web crawlers, often known as web robots or web spiders, are software applications that may download web pages more than once and automatically extract data or URLs as desired by users. In general, web crawlers are of two types, namely general web crawlers (GWC) and distributed web crawlers (DWC) depending on the number and operation of machines. DWC can then be categorized as multi-threaded or client-server. Figure 3 shows the process of web crawling data collection, which is repeated until all the data is collected [8].

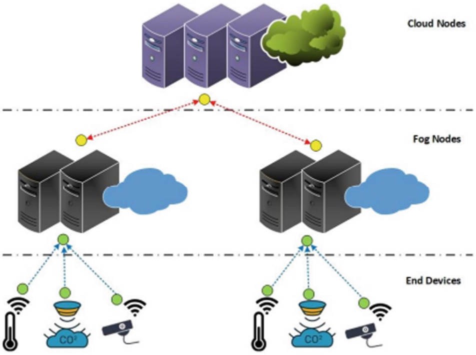

Fig. 4. Overview of fog computing architecture

2.3 Distributed Web Crawling

The distributed web crawler is divided into two parts, the first is multi-threaded (MT DWC), which makes various data threads make one in its entirety from either one machine, or server-client (SC-DWC), where several machines gather data concurrently. The second part is MT-DWC which has an overview like with SC DWC in one machine. The project load is divided into two, namely as follows: number one that works on the seed URL and one that makes one data taken from the website. The advantage of MT-DWC is that it is more economical to make new machines. Distributed web crawlers with a wide variety of machines use server-client designs to shorten data retrieval times [9].

2.4 Fog Computing

Fog computing introduces a layer between edge devices and the cloud. This layer relies on a group of small computing servers that are near the edge device and not necessarily on the device itself. Servers are connected and cloud servers are centralized, allowing for an intelligent flow of information [10]. These small units work together to handle data pre-processing, short-term storage, and rule-based real-time monitoring. Fog computing architecture reduces the amount of data transported through the system and improves overall efficiency [11]. An overview of the fog computing architecture is shown in Fig. 4.

2.5 Cloud Computing

Cloud computing is a network of several devices, computers, and servers that are connected via the Internet [12]. Cloud computing requires storage and access to enter data and programs via the internet from a computer (hardware). Users who use cloud computing do not have a structural infrastructure. This cloud computing realizes itself as a derivative of several other areas of computing. In the cloud market, there are 3 related parties in it. The three parties are as follows, End-user, Business Management, and Cloud service provider [13].

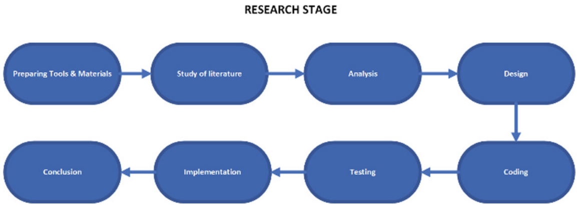

Fig. 5. Research stage

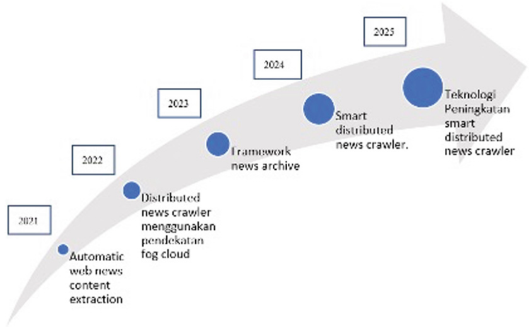
2021
2022
2023
2024
2025
Automatic web news content extraction
Distributed news crawler menggunakan pendekatan fog cloud
Framework news archive
Smart distributed news crawler.
Teknologi Peningkatan smart distributed news crawler

Fig. 6. Research road map

3 Methodology

The research design used in the development of the distributed news crawler is depicted in Fig. 5.

The study implemented a new web crawler approach that was considered more efficient in navigating In order to distribute URLs among concurrent crawler processes and prevent news building, URLs were organized by the domains used and divided into different prioritized URL queues. In particular, the proposed model can effectively utilize resources optimally in the cloud-fog layer by deploying a crawler distribution in the cloud-fog infrastructure to detect news. With fog clouds, the analysis is dynamically distributed across layers of fog and clouds allowing for real-time distribution. The research phase of distributed news crawlers starts from URL collection, URL filtering, scheduling, accessing URLs, and extracting news data. This research is focused on developing web crawlers to process distributed news crawlers [14]. In particular, the proposed model can effectively utilize resources optimally in the cloud-fog layer by deploying a crawler distribution in the cloud-fog infrastructure to detect news. With the fog cloud, analysis is dynamically distributed across the fog and cloud layers enabling real-time distribution.

This research was developed from a serial distribution process to a parallel distribution process to shorten the news search time. The following is the study's road map as shown Fig. 6.

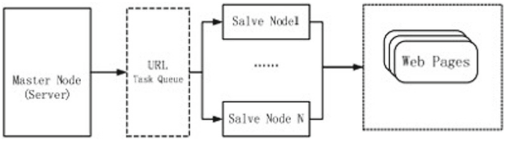

Fig. 7. Distributed web crawler system structure

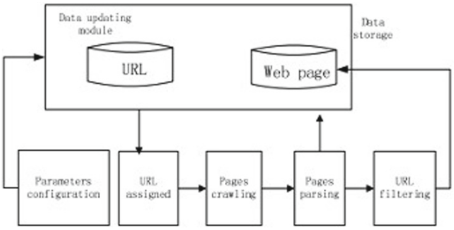

Fig. 8. Distributed web crawler system structure

4 Result

The research flow of the distributed web crawler compiled in this study is as shown in Fig. 7.

The workflow of this study is as follows: Programming by initializing crawler data, providing part of the cluster. Job assignment of slave node job tracker by the master node. Multi-threaded opening by the slave node, and after the task is received will be carried out to download the web page. Parsing web pages and storing Web material URLs. Adding URLs to the list that the crawler will assign after filtering and other operations. The division of tasks by the master nodes and see the status of each node. The section node monitors the communication performed with the works on these tasks as a master node via TaskTracker. JobTracker has the task of holding and managing all system communications, as well as allocating tasks. TaskTracker has the authority to work on the Map section and take on Tasks [15].

Combining with a combination of the system's business process solving, the distributed web crawler has five modules, namely the download page module, the page parsing module, the URL task allocation module, the parameter configuration module, the data update module, and the indexing module (if needed). More clarity distributed web crawler system structure can be seen in Fig. 8.

The uses and design of each standard are as follows:

a. Standard parameter configuration

- Function description: this standard has a function to perform a general extraction of system parameters. So that it can be monitored the flow of system performance. System input: Various parameters, web crawler link links, the amount of work, the form of files during the web crawler process, work paths, etc., need to be filtered. Result: null

b. Calculation Procedure: Create an XML file with the input parameters. URL-defined standards

   • Function details: Url text sharing by user and assignment to Mapper waiting for the web crawler. Input: Registration URL. Output: Sequence of partitioned <key, value> format URLs. Calculation process: A job task is created by the user software and sent to the cluster. Distinguish the URL text, then provide it to the Mapper. Every node executes this Map task.

c. Download page standards

   Algorithmic calculation: The segmentation algorithm has the main task of implementing the Hadoop input file format. The way it works is as follows. Calculating the size of each slice comes first, followed by calculating the total number of divisions. Consider it if it is smaller than a piece. Math.max (minSize, Math.min (goalSize, blockSize)) is the formula used to determine the size of each slice. Function specifics: Each node executes the crawler job after getting it. will have a row with several strategies. Then, do the web page download via multi-thread and do the storage in HDFS. Module download page. Formatted URL registration input. Websites are produced.

   Rearranging the waiting row comes first in the calculation process, after which a multi-thread is launched to obtain the website through the thread pool. Algorithmic calculation: In this study, the calculation algorithm will follow the replacement of the URL. In this study, researchers initially collected URLs after collecting the hostname from the list of URLs.

d. Page sharing standards

   Function details: Each node downloads the webpage, then must perform the elaboration of the page. The extraction activity of Links to URLs and content on the website should match what is needed. Web documents as input. Web posts and URL links are the output. Web pages are created using parser technology, and all URL links and other data are extracted from the pages. Both the web information and the URL link should be saved to the web page library. It's a Map Reduce operation.

e. Link filtering guidelines

   Function details: Before being included to the sequence, the link registration that is taken from the web page must first be normalized, filtered, and duplicated. Input: The link to which the extraction is performed. Output: links that have been filtered. Calculation process: The first step is, the researcher must correct the link to the standard link. The link filter is incorrect and non-compliant. Furthermore, to fit the strategy, meta information must be obtained and search for the robots file.txt first when web crawlers are carried out on web pages. Then filter the link with the parameters that have been set and duplicated. The last step will be added to the sequence. Algorithmic calculation: Perform duplicate link removal by passing key values during local deduplication.URL filtering module

f. Standard update data

   Function details: All nodes must be joined, and web page copy removal is performed, then a web page library update will be performed, and the links are filtered. Then it is duplicated to the center of the link. Input: Links and web pages from libraries of links and web pages. Output: Combines and optimizes URL links from web pages. Calculation procedure: allocating web documents and data links to each node after sharing them. Each node has algorithms that integrate and optimize its functions. Algorithmic Process: Performs the implementation of the algorithm on duplicated links. Next, a BitSet array is created and there is a partial hash function. Finally, the URL to BitSet mapping will be done with a hash function.

To create a web crawler that works properly according to expectations is to perform tasks simultaneously. From several web pages, it is required to distribute independently to provide the possibility of simultaneous access. Meanwhile, simultaneous distribution will reduce the cost of system transmission capacity assets. In general, web page crawlers are divided into three parts of the strategy that can be considered, namely as follows:

a. The first is the depth-first method, which is used when there are not too many pages to be downloaded.
b. The second strategy is the broad-first and best-first strategy. Due to the scattered nature of this web crawler, the initial search of the best and broadest is a useful search.

In this study, a new approach was designed for a smart web crawler system that is better distributed by navigating through links by making settings that are in accordance with the domain and dividing the link boundary into several URL queues that are prioritized, so that URLs can be distributed between web crawler processes to reduce the accumulation of the main material, namely news. In this study, link sharing is expected to facilitate the design of parallel crawlers by making it both error-proof by generating a balanced load distribution among all links on the remaining web crawlers, as well as scalable by sharing by domain and subdomain. Of course, the partition can hold the responsibility of collecting pages from the same domain. The system designed in this study could be further improved if all URLs' domain information could be obtained before obtaining linked Web pages.

We can know that in general, there are two types of divisions on web crawlers, namely:

1. Link-centric sharing: A link-centric share will collect various web crawlers to obtain duplicates of the same web page because the web page may be redirected to a different link. Web share architecture by way of assigning a link to a share. The copy of the link will be removed by the web crawler since the original link will be retrieved by the same web crawler process if there is a duplicate. A link is the type of resource that is transmitted between crawler process threads.
2. Sharing that is focused on content: This type of sharing specifies how it should be done after a web page has been retrieved and a link has been provided by a web crawler. In order to make changes to the multi-subject web page, the received page is further segmented into various small units of a single topic, which are then passed on to the appropriate crawler process.

In this study, the divisional technique involves combining the two theories to get rid of the accumulation of both links and material. The methodology used in this study provides unmistakable proof that detailed web pages are more likely to merge into sites linked to the domain they contain than randomly picked pages. In this study, a comparison between a single news crawler used to process news crawlers and distributed news crawlers using a distributed web fog cloud strategy will be made. This test is specifically designed to determine the proposed model's suitability for news detection in terms of duration, level of data gathering, speed, and data veracity.

5 Conclusion

The author employed a novel method to fog cloud in this study to create a distributed web crawler that is flexible, effective, load balancing, and scalable to find related posts on each news story. With this distributed web crawler, the author hopes to optimize news searches and help someone in finding news easily.

Considering the findings of the research that has been done, it can be said that the distributed web crawler uses a configurable, efficient, load balancing, and scalability fog cloud approach to determine related posts on each news, namely by using a crawler distribution. With the existence of a distributed web crawler, it can optimize news search and help someone in finding news easily. With multiple machines working together and multiple threads on each node, crawler systems can operate effectively. Distributed implementations of web crawlers have the characteristics of multipoint access, so they have greater total bandwidth and stronger processing capabilities. The next development in this study is the Framework news archive.

Authors' Contributions. Author did research, writing and also editing this manuscript.

References

1. Ahsan H T Y, Wibowo W C. A Fast Distributed Focused-web Crawling [J]. Procedia Engineering, 2014, 69(1):492–499
2. Shokouhi M, Chubak P, Raeesy Z. Enhancing Focused Crawling with Genetic Algorithms. Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05). 2005; 2: 503–508.
3. Abdillah, Fajri, Taufik Ichsan and Jumadi (2016). News Web Crawler Dengan Metode Xpath, Conference: Seminar Nasional Sains dan Teknologi 2015
4. Sawroop Kaur Bal, G. Geetha. (2016). Smart distributed web crawler, International Conference on Information Communication And Embedded System (ICICES 2016)
5. Pu Qiumei. (2016). The Design and Implementation of a High efficiency Distributed Web Crawler. 2016 IEEE 14th Intl Conf on Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress
6. Yong-Young Kim. (2019). Implementation of hybrid P2P networking distributed web crawler using AWS for smart work news big data, Peer-to-Peer Networking and Applications https://doi.org/10.1007/s12083-019-00841-0.
7. S. Lawrence, C.L. Giles,: Searching the World Wide Web. Science, Vol. 280, pp. 98–100, (1998) www.sciencemag.or
8. S. Chakrabarti, M. Van Den Berg, and B. Dom. Focused crawling: a new approach to topic-specific Web resource discovery. " Computer Networks, 31(11–16), pp.1623–1640, 1999

9. Ye Y, Ma F, Lu Y, et al. iSurfer: A Focused Web Crawler Based on Incremental Learning from Positive Samples [J]. Lecture Notes in Computer Science, 2004.
10. Lakhan, Mazin Abed Mohammed, Dheyaa Ahmed Ibrahim et al., Bio-inspired robotics enabled schemes in blockchain-fog-cloud assisted IoMT environment, Journal of King Saud University – Computer and Information Sciences, https://doi.org/10.1016/j.jksuci.2021.11.009
11. Zainudin, Ahmad, et al. (2021). Implementation of Fog Computing in Smart Home Applications Based on the Internet of Things Cess (Journal of Computer Engineering System and Science) p-ISSN: 2502-7131, Vol. 6 No. 1 January 2021
12. Wu M, Lai J. The Research and Implementation of Parallel Web Crawler in Cluster [J]. International Conference on Computational & Information Sciences, 2010:704–708.
13. Olston C, Najork M. Web Crawling [J]. Foundations & Trends in Information Retrieval, 2010, 4:175-246.
14. M. K. Hussein, M. H. Mousa, Efficient Task Offloading for IoT-Based Application in Fog Computing Using Ant Colony Optimization, IEEE Access, 2020
15. Gupta Sonali, Bhatia Komal Kumar. (2013). 2013 International Symposium on Computational and Business Intelligence.