End-to-End System For Data Crawling, Monitoring, And Analyzation Of E-Commerce Websites

Manh-Quang Do[1,*] , Thi Lan Nguyen[2], Dinh Duy Vu[2], Xuan Duc Tran[2], Thi-Quynh Nguyen[2], Ba-Nghien Nguyen[2], Van Tinh Nguyen[2], Ngoc-Anh Nguyen[3]

[1]Faculty Of Interdisciplinary Digital Technology (FIDT) PHENIKAA University
[2]Faculty of Information Technology, Hanoi University of Industry, Vietnam
[3]Giao Hang Tiet Kiem Joint Stock Company, Vietnam
*Corresponding author: Manh-Quang Do. Email: quang.domanh@phenikaa-uni.edu.vn;

Abstract

In an era of rapidly developing internet technology and increasing social requirements of individuals, creating and exploiting a large amount of data has become necessary for businesses to improve their competitiveness. Crawling data quickly and accurately to satisfy user needs has become crucial for data mining Many web crawling data methods have been utilized by businesses to acquire the desired information but crawling systems for e-commerce have not yet. In this paper, we proposed an end-to-end crawling system for e-commerce websites. We investigate the Lomart and Shopeefood websites and build a data collection system to analyze the retrieved data and then represent them in chart form to facilitate the most intuitive monitoring of the data crawling process. The source code and data are available at https://github.com/DoManhQuang/data_crawling.

Keywords: crawling system, e-commerce, data collection, web crawler, crawler intelligence, distributed crawlers

1 Introduction

Data collection [1,2] and analysis [3,4] play a crucial role in current economic and social development. Exploiting data from e-commerce websites and applications can help businesses better understand user behaviors, assess the effectiveness of marketing and advertising activities, and monitor and analyze competitors' movements, providing significant benefits. Data crawling [5,6,7] offers numerous advantages for current economic development, particularly with data from e-commerce platforms serving as the lifeblood of modern businesses. Data crawlers can be broadly classified into web crawlers [8,9,10], distributed crawlers [11-18], and crawler intelligence [19,20,21].

2. Related work

Firstly, the web-crawler method involves constructing a single-threaded data collection and analysis system. For example, Chunlin Li and Jingpan Bai [22] developed the world's first web crawler, Wanderer, in the Perl language at MIT. The paper proposed a web crawler model based on semantic analysis and spatial clustering to extract relevant content from web links. Yaning Yan and Jing Li [23] developed a network crawler and data analysis system using the Spring MVC framework. The system grabs data from specific sources to analyze television programs and video content from mainstream websites. Yani Ma et al. [24] proposed a method to analyze the structure of news websites, and design and implement a highly efficient simulated login model for data collection, which avoids the human authorization step in the login process. Sun Long and Li Yan proposed a Three-stage templated crawling technology for web data crawling. Experiments show that the use of technology can effectively crawl web pages retrieved from the database [25].

Secondly, the distributed crawler method involves building a data collection system using multi-threaded, distributed processing techniques, combined with Big Data platform technologies such as cloud, AWS, and Hadoop. For instance, Yong-Young Kim et al. [26] proposed a distributed web crawler using the master-slave crawler structure to improve the data collection efficiency of the web crawler. They proposed a hybrid P2P crawler that can collect web data using the cloud service platform provided by Amazon Web Services (AWS). The hybrid P2P networking distributed web crawler using AWS (HP2PNC-AWS) is applied to collect news on Korea’s current smart work lifestyle from three portal sites. A P2P crawler described in [27,28] introduces the architecture of a distributed Hadoop platform and proposes a design scheme for a distributed web crawler based on Hadoop and P2P technology. The approach is to partition the RIA model resulting from the crawling process across multiple storage devices in a P2P network, making the distributed data structure invulnerable to single points of failure. Xuejiao Ren, Hairong Wang, and Diwei Dai [29] proposed “A Summary of Research on Web Data Acquisition Methods Based on Distributed Crawler.” This paper tracks and studies the latest technical methods and introduces distributed web crawler technology. Based on this, the distributed crawler methods and strategies based on Scrapy-Redis, cloud platforms, and Nutch are analyzed. The results show that the use of distributed crawlers has obvious advantages in obtaining large-scale Web data.

Lastly, the crawler intelligence [30] method involves building an intelligent data collection system using machine learning and optimization techniques. For example, Mehdi Assefi et al. [31] proposed “An Intelligent Data-Centric Web Crawler Service for API Corpus Construction at Scale.” In this paper, they introduce a data-centric web crawler service to collect, analyze, and construct a large corpus of API documentation. The proposed API web crawler intelligently harvests more than 2.8M

API documentation pages, using a machine-learning-based approach with an accuracy of 91.32% to select only web API pages (REST). They also conducted an extensive end-to-end real-world evaluation, where the proposed API web crawler not only collects a large number of API pages but also successfully validates 1,222 APIs out of 1,521 target APIs with a success rate of 80.34%. Ni Putu Linda Santiari et al. [32] proposed research that uses a crawler framework to explore web and deep content, store data in a database, and classify the website levels using the fuzzy-KNN method. Initial findings indicate the web content comprises around 20% surface web, 7.5% web bergie, 20% deep web, 22.5% charter, and 30% dark web. However, the research is limited, and more data is needed to improve the results.

In this paper, we proposed a multi-threaded data collection, monitoring, and analysis system for e-commerce websites. Implemented real-world evaluation on two e-commerce platforms, Shopee Food, and Lomart. The collected data is in non-relational format and stored in MongoDB, and analysis is performed using Grafana. Our main contributions to this paper are as follows:

- Proposed a new method for collecting all data from the e-commerce platform.
- Implementing a data collection system and storage in MongoDB due to its high compatibility with unstructured data.
- Implementing REST API for a data monitoring and analysis system using Grafana.

3. Methodology

3.1 Proposed System Model

Our proposed information-gathering system retrieves data from websites using API as Fig. 1. The API data source returns a vast amount of data, therefore, our system aims to fetch the necessary data quickly, accurately, and according to practical needs. Our data collection system uses API to get information from Shopee-food and Lomart websites. The steps to get API from the website to the crawl system are shown in Figure 1 below. 1) Press F12 to open the source code of the website to get the API of Lomart and Shopee-food. 2) Click Network => Fetch/XHR, then press F5 to display the APIs of the website. At this time, the APIs of the website are displayed. 3) Click on the API you want to get, then click Preview to view the API data to be retrieved.

As Figure 2 shows, our data collection system crawls the website data in five steps: 1) Browse the cities, 2) Browse the main product categories, 3) Browse the list of pages containing products, 4) Browse the list of products on a page, 5) Check the data and store it in the database. This cycle repeats until all the website data is crawled.

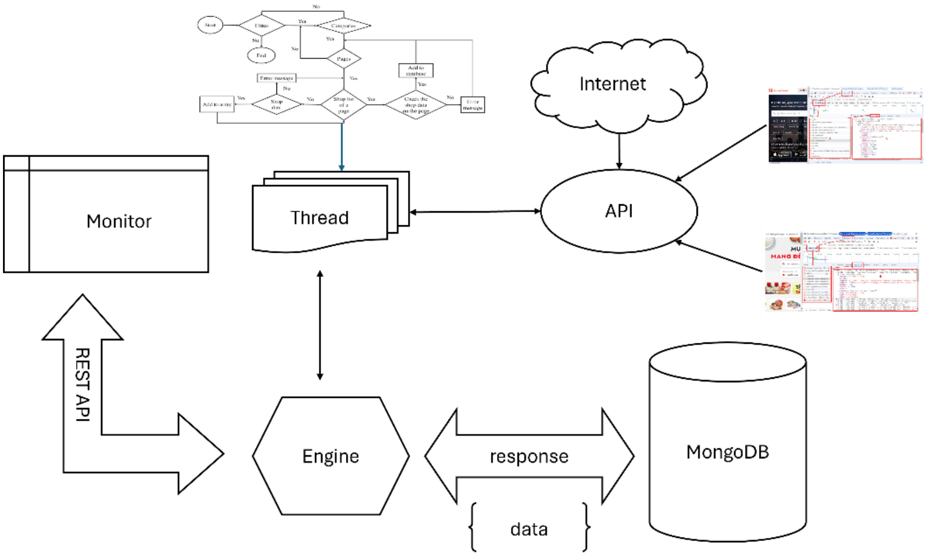


Fig. 1. A diagram of a process flow

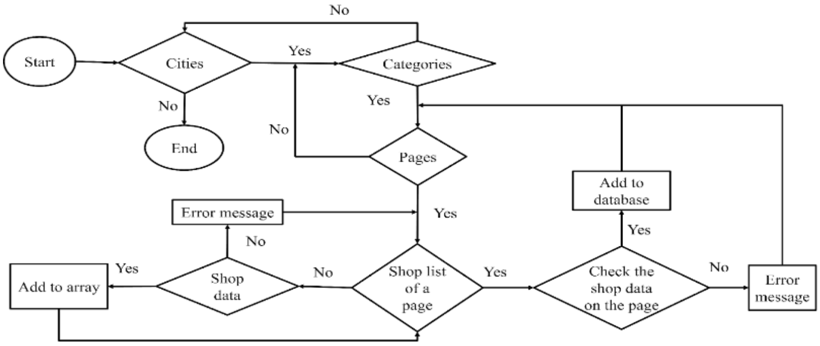


Fig. 2. The process of web crawling

3.2 Crawl Strategy Analysis

The system collects and browses the data in a hierarchical tree structure after accessing the API. In Figure 3 below, the data is browsed sequentially from Level 1 (the system browses each city and product category) to Level 2 (browsing to the subcategories of the current product category) down to Level 3 (browsing to the stores) and finally Level N (at this point, the product information and store information are fully collected). This process is repeated until all the data on the website is collected.

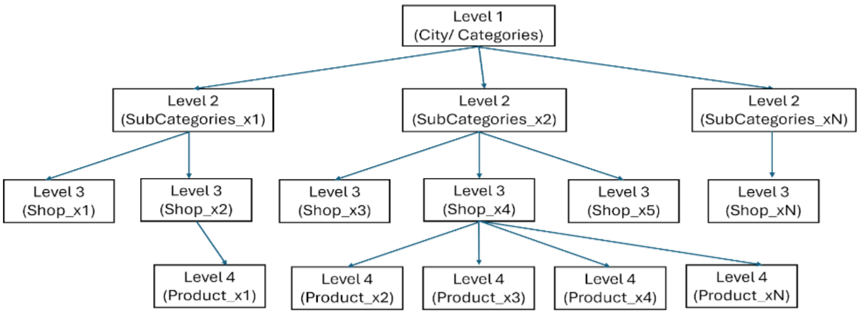


Fig. 3. Data collection hierarchy

3.3 Multi-Threaded Processing

With a large amount of data collected from the system, processing data incrementally takes a lot of time and consumes a lot of computer resources. Therefore, to increase the processing speed, utilize computer resources, and ensure good response and interaction during the task, we have added multi-threaded processing to the system. In Figure 4 below, we describe the multi-threaded data processing process, the cities are divided into smaller groups, each group corresponding to a data processing stream of the system.

For example, if there are 20 cities and 5 processing streams, there are 4 cities in each stream. This multi-threaded processing helps the system retrieve data quickly, accurately, and efficiently while saving resources. Once the data has been collected, we use Fast REST API to build a web API, then display them in the form of charts and tables using Grafana to visualize the data.

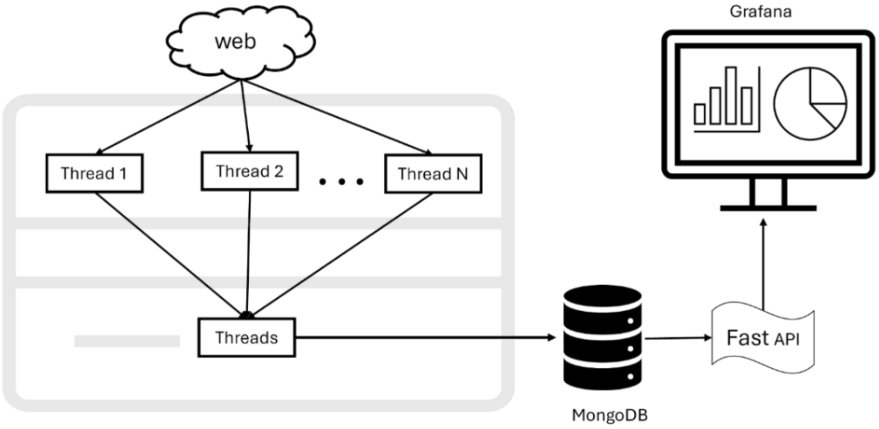


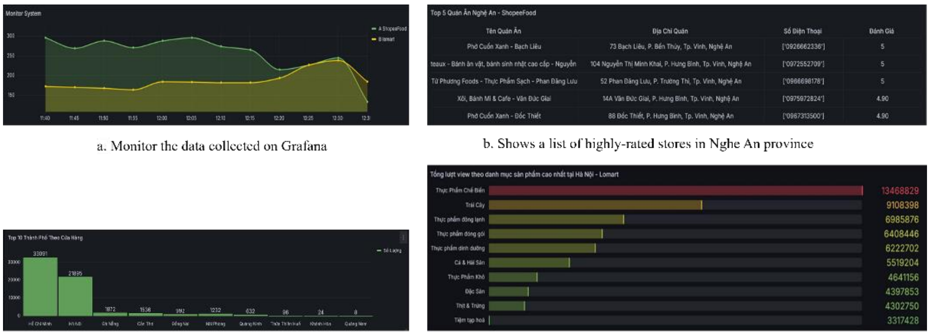
Fig. 4. Multithreaded distributed web crawler

4. Experimental Results

After the experimental results, the system crawled nearly 80,000 data points of selling shops from the two e-commerce platforms, Lomart and Shopee Food. Below is a demo of some of the experimental results from our system.

4.1 Data Analyzation

The amount of data on Lomart and Shopee Food is substantial. Therefore, we have built a system to help track the time of information collection. In Figure 5, we have built a crawling system for monitoring and analyzation data collected.



a. Monitor the data collected on Grafana

Top 5 Quán Ăn Nghệ An - ShopeeFood

Tên Quán Ăn	Địa Chỉ Quán	Số Điện Thoại	Đánh Giá
Phở Cuốn Xanh - Bạch Liêu	73 Bạch Liêu, P. Bến Thủy, Tp. Vinh, Nghệ An	['0926662336']	5
teaux - Bánh ăn vặt, bánh sinh nhật cao cấp - Nguyễn	104 Nguyễn Thị Minh Khai, P. Hưng Bình, Tp. Vinh, Nghệ An	['0972552709']	5
Từ Phương Foods - Thực Phẩm Sạch - Phan Đăng Lưu	52 Phan Đăng Lưu, P. Trường Thi, Tp. Vinh, Nghệ An	['0986698178']	5
Xôi, Bánh Mì & Cafe - Văn Đức Giai	14A Văn Đức Giai, P. Hưng Bình, Tp. Vinh, Nghệ An	['0975972824']	4.90
Phở Cuốn Xanh - Đốc Thiết	88 Đốc Thiết, P. Hưng Bình, Tp. Vinh, Nghệ An	['0987313500']	4.90

b. Shows a list of highly-rated stores in Nghe An province

c. The list of the number of shops in provinces/cities

Tổng lượt view theo danh mục sản phẩm cao nhất tại Hà Nội - Lomart
Thực Phẩm Chế Biến 13468829
Trái Cây 9108398
Thực phẩm đông lạnh 6985876
Thực phẩm đóng gói 6408446
Thực phẩm dinh dưỡng 6222702
Cá & Hải Sản 5519204
Thực Phẩm Khô 4641156
Đặc Sản 4397853
Thịt & Trứng 4302750
Tiệm tạp hoá 3317428

d. The number of views of users on each product category

Fig. 5. Dashboard monitor and analysis for data collected

4.2 Crawling Speed

The results evaluated in Table 1 demonstrate the data collection speed of the system using multi-thread processing. The amount of retrieved data increases linearly. The longer the time, the more data is obtained, the higher the number of threads, and the quicker the data collection. “SF” is ShopeeFood, and “T” is Threads.

Table 1. The data collection speed of the system using multi-thread processing.

Time (minutes)	SF	Lomart	SF	Lomart	SF	Lomart	SF	Lomart
	1T		2T		4T		8T	
5	200	72	1012	528	1600	624	491	1152
15	690	436	1384	1744	3756	2008	3248	3752
30	1323	1096	3980	3404	6352	4384	4401	6352
60	2347	2385	4406	4904	9412	8480	7022	13456

However, up to a certain number of threads, such as Shopee Food, the data collected by 8 threads is less than 4 threads. This is because, with a sufficiently large system, all the data on Shopee Food can be obtained quickly, causing Shopee Food to restrict the number of requests from the system to prevent potential attacks. In contrast, Lomart's system remains secure even with the requested number of threads, allowing the data amount to continue increasing over time along with the number of threads. Based on this evaluation table, we select a multi-thread configuration for some data collection processes to prevent blocking from websites.

5. Conclusion and Discussions

Data collection and analysis play a vital role in today's economic and social development. Through these numbers and data, experts and businesses can gain a clear understanding of market trends, current consumer needs, and competitors, enabling informed business decisions. This paper introduces a multi-threaded system for data collection, monitoring, and analysis on e-commerce websites. Real-world evaluations were conducted on two platforms, Shopee Food, and Lomart. The collected data is stored in MongoDB in a non-relational format. We also built a REST API for monitoring the data crawling progress using charts and tables, making it easy for everyone to observe the data visually on Grafana.

As the volume of data increases, simplifying and enhancing its applicability becomes essential. The next step involves optimizing the data by building an AI training dataset to assess its quality, generality, and applicability.

References

[1]. 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.

[2]. Maheshwar, Poonam. 2016. "A Cloud-based Web Crawler Architecture". International Journal of Engineering and Management Research. Volume6. Page 148-152

[3]. H. Wu, F. Liu, L. Zhao and Y. Shao, "Data Analysis and Crawler Application Implementation Based on Python," 2020 International Conference on Com-puter Network, Electronic and Automation (ICCNEA), Xi'an, China, 2020, pp. 389-393, doi: 10.1109/ICCNEA50255.2020.00086.

[4]. 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.

[5]. ElAraby M E, Moftah H M, Abuelenin S M et al, "Elastic Web Crawler Ser-vice-Oriented Architecture Over Cloud Computing," Arabian Journal for Science and Engineering, vol. 43(12), 2018, pp. 8111-8126.

[6]. Liu J, Li F, Jiang S, "Annealing crawler algorithm for storm disaster theme based on comprehensive priority and host information," Computer Sciencee, vol. 46(02), 2019, pp. 215-222.

[7]. Lin S, Yuan Z, Li X, "Semantic focusing crawler method combined with text density," Computer Applications and Software, vol. 36(09), 2019, pp. 270-275.

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

[9]. Zeng J, Zhang Y, Zheng J, Huang G, Chen R, “Multi-data source oriented web crawler implementation technology and application,” Computer Science, vol. 46(05), 2019, pp. 304-309

[10]. Farooq B, Husain M S, Suaib M, “CRAWLING OF JAPANESE REAL-ESTATE WEBSITES USING SCRAPY,” International Journal of Advanced Research in Computer Science, vol. 9(Special Issue 2), 2018, pp. 64-67

[11]. Kim Y Y , Kim Y K , Kim D S , et al, “Implementation of hybrid P2P net-working distributed web crawler using AWS for smart work news big data,” vol. 13(2), 2020, pp. 659-670.

[12]. Zhang S, Tan H, Chen L, lv B, “Uni Crawl: An efficient load balancing strat-egy for distributed crawler systems,” Computer Engineering, vol. 45(11) , 2019, pp. 62-67.

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

[14]. Dong Y, Yang L, Ma X, “Research on Active Acquisition Distributed Web Crawler Cluster,” Computer Science, vol. 45(S1), 2018, pp. 428-432.

[15]. Ma L, Feng X, Dou Y, Gao T, Zhu R, Wu Y, “Research and Implementation of Distributed Crawler,” Computer Technology and Development, vol. 30(02) , 2020, pp. 192-196.

[16]. Wang B, “Design and Implementation of Distributed Crawler System Based on Scrapy Framework,” Proc. Hefei University of Technology, 2019.

[17]. Fang Q, Cheng Y, “Design and Implementation of Distributed Crawler Based on Docker Container,” Electronic Design Engineering, vol. 28(08), 2020, pp. 61-65.

[18]. DingY, “Design and Implementation of Vertical Search Engine Based on Web Crawler,” Proc. Guizhou University, 2019.

[19]. Xiao J, “Theme crawler based on improved VIPS algorithm and improved gray wolf optimization algorithm,” Proc. East China Normal University, 2019.

[20]. Hou M, Cui Y, Hu J, “Research on Intelligent Crawling Algorithm Based on Reptiles,” Computer Applications and Software, vol. 35(11), 2018, pp. 215-219,277.

[21]. Liu J, Li F, Jiang S, “Annealing crawler algorithm for storm disaster theme based on comprehensive priority and host information,” Computer Sciencee, vol. 46(02), 2019, pp. 215-222.

[22]. Chunlin Li, Jingpan Bai: Automatic content extraction and time-aware topic clustering for large-scale social network on cloud platform (2018).

[23]. Y. Yan and J. Li, "Design and Development of an Intelligent Network Crawler System," 2018 2nd IEEE Advanced Information Manage-ment,Communicates,Electronic and Automation Control Conference (IMCEC), Xi'an, China, 2018, pp. 2667-2670, doi: 10.1109/IMCEC.2018.8469619.

[24]. Y. Ma, Q. Song and Y. Fu, "Simulated website login to improve network data crawling efficiency," 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chengdu, China, 2019, pp. 1722-1726, doi: 10.1109/IAEAC47372.2019.8997938.

[25]. S. Long and L. Yan, "Crawling Deep Web Data Based on Three-stage Template," 2022 7th International Conference on Big Data Analytics (ICBDA), Guangzhou, China, 2022, pp. 30-34, doi: 10.1109/ICBDA55095.2022.9760358.

[26]. Yong-Young Kim, Yong-Ki Kim, Dae-Sik Kim, Mi-Hye Kim: Implementation of hybrid P2P networking distributed web crawler using AWS for smart work news big data (2019).

[27]. Xu H, Li K, Fan G (2018) An Improved Strategy of Distributed Network Crawler Based on Hadoop and P2P: Applications and Techniques in Cyber Security and Intelligence, In book: International Conference on Applications and Techniques in Cyber Security and Intelligence ATCI 2018 (pp.849-855)

[28]. Khaled Ben Hafaiedh, Gregor V. Bochmann, Guy-Vincent Jourdan, Iosif-Viorel Onut: Fault Tolerant P2P RIA Crawling, International Conference on Networked Systems (pp. 32–47).

[29]. X. Ren, H. Wang and D. Dai, "A Summary of Research on Web Data Acquisition Methods Based on Distributed Crawler," 2020 IEEE 6th International Conference on Computer and Communications (ICCC), Chengdu, China, 2020, pp. 1682-1688, doi: 10.1109/ICCC51575.2020.9345157.

[30]. A. Darshakar, "Crawler intelligence with machine learning and Data Mining integration," 2015 International Conference on Pervasive Computing (ICPC), Pune, India, 2015, pp. 1-6, doi: 10.1109/PERVASIVE.2015.7087203.

[31]. M. Assefi et al., "An Intelligent Data-Centric Web Crawler Service for API Corpus Construction at Scale," 2022 IEEE International Conference on Web Services (ICWS), Barcelona, Spain, 2022, pp. 385-390, doi: 10.1109/ICWS55610.2022.00064.

[32]. I. G. S. Rahayuda and N. P. L. Santiari, "Crawling and cluster hidden web using crawler framework and fuzzy-KNN," 2017 5th International Conference on Cyber and IT Service Management (CITSM), Denpasar, Indonesia, 2017, pp. 1-7, doi: 10.1109/CITSM.2017.8089225.