International Journal on Science and Technology (IJSAT)
E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

From Web to File: Creating a Scraper for Structured E-commerce Product Data

Manavlal Nagdev[1], Vineeta Rathore[2], Maya Baniya[3], Md Muaviya Ansari[4], Mustafa Sultan[5]

[1,2,3,4,5]Department of Engineering
Medicaps University
Indore, India
[1]manav.n2705@gmail.com,[2]vineeta.rathore1@gmail.com,[3]mayayadav55@gmail.com
[4]muaviyaansari57@gmail.com,[5]mustafasultan5250@gmail.com

Abstract

The acquisition of organized product data continues to be a crucial obstacle in the dynamic world of e-commerce. This problem is made worse by the growing complexity of contemporary websites, which include dynamic content and anti-scraping features. By addressing the shortcomings of current approaches, this paper offers a thorough methodology for creating a reliable web scraper designed especially for Indian e-commerce platforms. To efficiently handle static as well as dynamic material, the suggested approach incorporates Beautiful Soup and Selenium with Flask and React.js. Overcoming anti-scraping mechanisms, guaranteeing data accuracy through sophisticated preprocessing approaches, and offering actionable insights through data visualization are some of the research's main accomplishments. This study also includes scalability to manage big datasets across various e-commerce platforms, ethical scraping methods, and compliance with robots.txt instructions. The scraper's ability to extract, clean, and analyze data is confirmed by experimental findings, providing a scalable and morally sound option for automated e-commerce data extraction.

Keywords: Web scraping, e-commerce, data preprocessing, Selenium, Beautiful Soup, data visualization, anti-scraping techniques, scalability, ethical scraping.

1. INTRODUCTION

Over the past ten years, the e-commerce industry has grown at an unprecedented rate due to technological advancements, widespread internet access, and changing consumer behavior. Platforms like Amazon, Flipkart, Myntra, and Ajio have transformed the retail landscape by providing consumers with a wide range of options and unparalleled convenience, but this rapid evolution has also made it imperative for businesses to use data-driven insights to adapt to a competitive environment. Accurate and structured product data is now a crucial asset that informs decisions about pricing strategies, inventory management, marketing campaigns, and customer engagement.

Finding organized and useful information is still a difficult task, even with the wealth of data on e-commerce platforms. E-commerce websites use advanced anti-scraping techniques, rely extensively on

JavaScript to render dynamic content, and regularly change their architecture. These traits present serious challenges for conventional data collection techniques. Businesses and researchers looking to evaluate market trends or obtain a competitive edge cannot afford to use manual data extraction methods because they are laborious and prone to human error.

One effective way to deal with these issues is through web scraping. Large amounts of information can be gathered more accurately and efficiently by automating the process of extracting data from websites. However, current web scraping solutions often fall short when applied to modern e-commerce platforms. Many fail to effectively process dynamic content, circumvent anti-scraping measures, or scale up to meet the demands of large-scale operations.

The incapacity of current scraping methods to handle dynamic content is one of their main drawbacks. Because JavaScript is not included in the original HTML source code, static scraping technologies are unable to access the dynamic content generation and rendering capabilities of modern e-commerce websites. Automated data extraction is made more difficult by anti-scraping methods used by these platforms, such as rate limitation, IP banning, CAPTCHA verification, and user-agent identification. Furthermore, a lot of preprocessing is required to make the retrieved data appropriate for analysis because it is frequently unstructured, inconsistent, and full of unnecessary information. Another issue is scalability, since many scraping technologies are unable to effectively manage enormous datasets, which results in bottlenecks in server speed, processing time, and memory utilization.

This work presents a sophisticated web scraping system specifically for Indian e-commerce systems in order to address these challenges. Supporting both dynamic and static content, the system runs modern technologies including Selenium and Beautiful Soup for effective data extraction and processing and Flask and React.js for backend and frontend operations respectively. By following robots.txt rules, restricting request rates, and avoiding unnecessary burden on target servers, the system also conforms with ethical scraping criteria.

Overcoming these challenges provides a scalable, moral, and efficient approach to extract structured e-commerce data under suggested system. This paper focuses on the design, execution, and performance assessment of the system as well as underlines its ability to offer insightful analysis in a very competitive environment.

2. Literature Review

With methods ranging from DOM parsing to sophisticated crawling frameworks, web scraping has been well studied. While tools like Scrapy concentrate on scalability for big datasets [1], UzunExt's effective string-matching techniques stress computational efficiency [2]. Many of these techniques lack flexibility to accommodate dynamic content and fail to incorporate real-time user feedback, notwithstanding their strengths. This restriction is especially important since JavaScript-generated web pages are now the main source of dynamic, user-specific content on contemporary e-commerce systems. Static parsers thus sometimes overlook important data, so compromising the completeness and dependability of the obtained knowledge.

Frameworks like Selenium and Puppeteer have helped to solve the challenges presented by dynamic content. Selenium can be used to scrape websites heavy in JavaScript since it replics user interactions with online pages. Though Selenium has great capacity for automating online interactions, its processing load is

more than that of lightweight parsers like Beautiful Soup. Beautiful Soup struggles with dynamic content and AJAX calls but performs effectively for stationary web pages since it is simple and efficient. Recent studies indicate that combining Beautiful Soup for parsing stationary HTML elements with Selenium for JavaScript rendering offers a balanced approach of managing several content kinds. [3][4].

Website anti-scraping features like IP filtering, rate limitation, and CAPTCHAs add another level of difficulty. Proxy servers and user-agent spoofing are frequently used to circumvent these restrictions. Proxy rotation reduces the likelihood of discovery and blockage by making sure that requests originate from different IP addresses. However, some advanced anti-scraping techniques, such as JavaScript-based issues and device fingerprinting, require more sophisticated solutions. It has also been investigated to use CAPTCHA-solving services to get past automated obstacles, but these methods raise ethical and legal concerns regarding compliance with website rules. [5].[6].

Machine learning has drawn interest as a potentially useful technology for enhancing web scraping methods. The efficiency and accuracy of the scraping process can be improved by using classification algorithms to find patterns in the scraped data. Customer reviews and other unstructured data are increasingly being parsed using natural language processing (NLP) techniques to produce insights that may be put to use. Convolutional neural networks (CNNs) have been used in image-based scraping techniques to extract visual components from e-commerce sites, such as product photos and ads. These techniques have the promise, but their real-time applicability is limited by their high computing resource and annotated dataset requirements [7][8].

An additional different approach is employing heuristic-based systems capable of detection and adapting to changes in web profile topologies. Heuristics can detect and traverse dynamically loaded parts, but they are limited in responding to quickly changing web design, as they rely on pre-rules. These systems are also still limited in terms of scalability, especially for websites that use different layouts or have different types of content. Adopting heuristic models in conjunction with machine learning models has shown some potential for overcoming these obstacles, but further development is needed to improve effectiveness [9].

Current methodologies lack the robustness of pipelines to clean and transform raw data into standardized formats. Data cleaning enhances usability of the extracted data through the correction of errors such as duplicates and missing values. Deduplication, standardization, and transforming data into a structured format such as CSV or JSON, are all important aspects of preparing data to be useful. Research indicates how relevant it is to directly integrate these pipelines within scraping systems to optimize their usefulness [10] [11].

Scalability of web scraping is still a major challenge. Many present systems find it difficult to manage several requests at once, which lowers output and results in lag in response times. Distributed systems like those developed with Scrapy can scale cleaning chores among several nodes. These systems restrict their usability for non-technical people, though, since they sometimes need major infrastructure and setup [12] [13].

Although recent studies show that web scraping methods have considerably advanced, there are still many issues. For many tools, processing dynamic material, including real-time user interaction, and visualizing data remain difficult. Technical debates sometimes ignore ethical issues in scraping techniques, including respect of terms of service and privacy laws. Closing these gaps calls for an interdisciplinary strategy considering ethical standards and technological developments. [14] [15].

3. PROPOSED WORK

The suggested project consists in building a thorough web scraping system designed especially to solve the problems presented by contemporary e-commerce systems. This part clarifies the goals, approach, and special characteristics of the system.

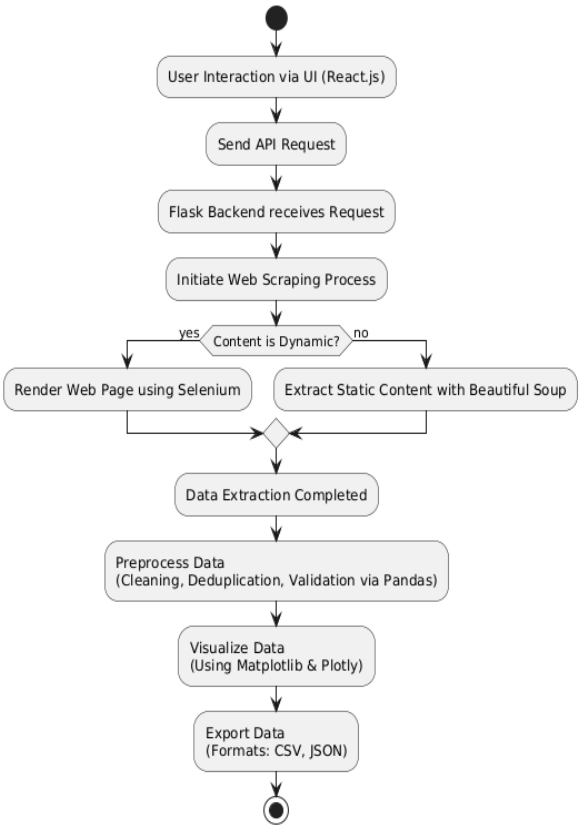
A. Objectives

The main purpose of this research is to develop a scalable and dynamic web scraping framework. The efficient extraction of data from web pages utilizing primarily javascript to render content is among the most important goals of the system. The design will ensure the framework can extract large amounts of data while maintaining accuracy and consistency through robust preprocessing techniques. While ensuring effective data management, the system also highly prioritizes ethical web scraping practices, such as following robots.txt protocols and establishing a request throttling mechanism. The system will also aim to provide actionable insights through advanced visualizations and exportable functionality to both CSV and JSON file formats.

B. Methodology

The architecture of the system is modular, separating its frontend and backend systems. React.js provides a user-friendly interface for the frontend to populate scraping parameters. Meanwhile, the backend uses the Flask framework to process data, support scraping logic, and expose API endpoints. This division of function can enhance resiliency and support maintainability.

By utilizing the visualization and export functionalities offered by some libraries such as Matplotlib and Plotly, users can create visual insights about aspects like product availability and price trends. The solution also facilitates exporting clean data in well-known formats like CSV and JSON for further analysis. Issues with scalability were overcome by utilizing a combination of multi-threading and asynchronous I/O operations, which efficiently handle resource allocation and allow multiple scraping operations to be performed simultaneously without performance lagging. Given the system ensures robots.txt compliance and automates rate limiting queries to reduce server burden, this strategy is founded on ethical compliance.



II. Results and Analysis

In order to assess the capabilities of the web scraping software that was developed, a test scrape was conducted on washing machine product listings from the three most popular e-commerce sites in India. The data extracted from the site produced insights into product descriptions, pricing, customer ratings, and discounting. Visualizations derived from the test have been created and are exhibited in Figures 1 to 4.

Fig.1 Word cloud

The word cloud represented in Figure 1 illustrates the most common words present in product titles; some examples include "Washing Machine," "Front Load," "Fully Automatic," and "Top Load." This knowledge can help e-commerce businesses optimize product descriptions for search visibility and product discovery.

Fig.2 Histogram

Figure 2 presents a histogram of the distribution of product prices. Most washing machines are found in the ₹15,000-₹30,000 price range, with availability dropping sharply beyond ₹60,000. This tells us that consumers prefer to purchase mid-range washing machines. The data implies that the higher price point is for a more niche audience, whereas the consumer overwhelmingly opts for affordable and value-pricing models.

Fig.3 Scatter Plot

Figure 3's scatter plot looks at discounted price and customer rating. There appears to be a cluster of highly rated goods, those rating >4, in the ₹10,000 - ₹40,000 price range, suggesting that price matter for affordability contributes to customer satisfaction. Moreover, high ratings were observed in all price segments which may be partially explained by factors such as brand reputation and product characteristics or features. The study shows customers will pay a premium for a dependable, well-reviewed product. However, we still find evidence that price and affordability are the most significant factor in purchasing behavior.

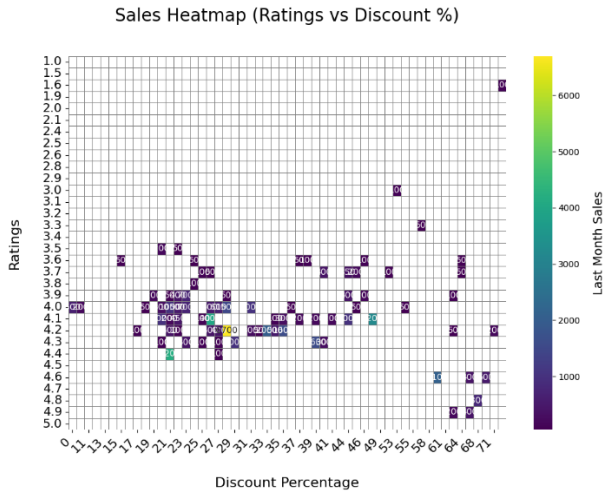


Fig.4 Heatmap

The heatmap presented in Figure 4 displays the affiliation between discount percentage and ratings. A cluster observed in the 10% to 40% discount range does indicate that customers commonly react positively to moderate discounts. Very large discounts, those greater than 50%, do not reflect greater rating scores, indicative of some concern to the level of product quality or reliability. This study emphasizes that price balance is an important aspect of product pricing. In this case discounts lure customers, but do not create a discounted perception of the value associated with a product.

Moreover, users can either download the extracted data in CSV and JSON formats or use the the visualizations interactively. The buttons in the interface are highlighted in Figures 5 and 6.

Fig.5 Download Buttons

Fig.6 Download Visualizations

The outcome of this test case validates the scraper's capability to effectively extract structured e-commerce data in a manner suitable for market trend tracking in real-time. The data obtained from the washing machine listings suggest that mid-tier products dominate the market because they offer more affordable pricing options for consumers. Also, while discounts can assist in affecting consumers' perceptions, excessive markdowns can result in consumer doubts about the durability and trustworthiness of the product. The insights from the washing machine data set demonstrate the scraper's efficacy of producing insights that businesses can use for intelligence, insights that would be relevant for e-commerce businesses looking to adjust pricing and product listings.

4. Conclusion

This paper presents a powerful web scraping framework to address the limitations of current e-commerce platforms. By incorporating contemporary web scraping technologies (Beautiful Soup, Flask, React.js, and Selenium), the system deals with dynamic content, circumvents anti scraping measures, and provides valid and structured data for analysis. The framework is guaranteed to be applicable to large datasets due to its self-scalable architecture, while its compliance with operational and legal guidelines is protected due to the framework's ethical considerations.

Among some of the important contributions that the proposed system brings to the field is the ability to extract information from content rich in JavaScript, process that information appropriately, and present results that can inform action with advanced visualization techniques. These features make the system a valuable resource for businesses interested in leveraging e-commerce data for a sustainable advantage and making informed business decisions. The successful testing of the system across several platforms supports its potential to offer a scalable and ethical approach to automated extraction of e-commerce data.

5. Future Scope

To advance the area of business forecasting, as well as customer decision-making, the future of web scraping may begin to involve machine learning models to predict price trends or suggest the best time to buy. Researchers may also look into advanced anti-scraping technologies, like browser fingerprinting, proxy rotation, and advanced CAPTCHA solvers, to improve the resilience of data extraction.

There are expected scalability benefits to additional domains, improved cloud-based storage solutions for handling large data sets, and plans to use distributed scraping frameworks that can efficiently manage requests being sent. In addition, the creation of a mobile-friendly interface is designed to create a responsive mobile application that will allow even more users to access data in real-time, as well as allow users to initiate scraping functionality when they are away from their desktop computers.

Another intriguing strategy could additionally be an extra API integration that would empower businesses to seamlessly integrate the scraper's capabilities into their operations. With real-time alerts, users would receive timely alerts on key changes in pricing or stock status for specific products. The addition of more sophisticated data analytics could also develop fully-fledged analytics dashboards that

offer prescriptive/predictive insights from the scraped data (e.g. market demand trends, product popularity indices).

REFERENCES

1. Lü et al., “A Survey on Web Scraping Techniques,” Journal of Data and Information Quality, 2016.
2. Uzun Erdinç, “Web Scraping Advancements,” IEEE, 2020.
3. Ryan Mitchell, “Web Scraping with Python: Collecting More Data from the Modern Web,” O’Reilly Media, 2018.
4. Bright Data, “Comprehensive Web Scraping Guide,” 2025.
5. Richard Lawson, “Web Scraping for Dummies,” Wiley, 2015.
6. Faizan Raza Sheikh et al., “Price Comparison using Web-scraping and Data Analysis,” IJARSCT, 2023.
7. PromptCloud, “How to Scrape an E-commerce Website,” 2024.
8. ScrapeHero, “Data Extraction for E-commerce Platforms,” 2024.
9. Aditi Chandekar et al., “Data Visualization Techniques in E-commerce,” IJARSCT, 2023.
10. Google Developers, “Advanced Web Scraping Techniques,” 2025.
11. Mitchell, R., “Modern Web Scraping Practices,” ACM Digital Library, 2023.
12. Bright Data, “Guide to E-commerce Web Scraping,” 2025.
13. Shreya Upadhyay et al., “Articulating the Construction of a Web Scraper for Massive Data Extraction,” IEEE, 2017.
14. Sandeep Shreekumar et al., “Importance of Web Scraping in E-commerce Business,” NCRD, 2022.
15. Niranjan Krishna et al., “A Study on Web Scraping,” IJERCSE, 2022.
16. Vidhi Singrodia et al., “A Review on Web Scraping and its Applications,” IEEE, 2019.
17. Aditi Chandekar et al., “The Role of Visualization in E-commerce Data Analysis,” IJERCSE, 2024.