Web Content Mining: A Survey of Methods, Tools, and Challenges through Machine learning Techniques

Kashish[1] ,Satyam vishwakarma[2] ,Prakhar[3],Arjun Singh[4],Dr.Anuranjan Mishra[5],

Arun Kumar Singh[6]

[1,2,3,4,5] Department of Computer Science and Engineering

[1,2,3,4,5] Greater Noida Institute of Technology Greater Noida,India

Abstract— Web scrapper for product price analysis is designed to compare the price of goods and services from a range of various providers and multiple platforms. This will serve the consumers to find the required product under the best price range. Web scrapper for product price analysis will help the customer in saving their money by making decision to choose products through online at cheap prices. In today's busy lifestyle, most of the customers prefer to buy products online because it helps them to save their time. Web Scraper will help gradually to save time as it collects the data from various shopping platforms and show the price, ratings and reviews of the same product at once which helps the customers to buy the product according to their budget. As this is the price comparison website, so customers don't have to travel store to store to buy a product at cheap price. Consumers just can check the price of the product provided at different shopping platforms and compare them easily. The name of this project is Web Scrapper for Product Price Analysis for the price comparison of products to provide easy buy services for the consumers. Web scrapper for product price analysis helps to find great deals for the customers and the best deals are clearly highlighted at the top. Web scraper for product price analysis uses web scrapping techniques to obtain the best deals and to fetch the detailed information of the same product at different shopping platforms. In this way, this paper aims to provide solutions for the consumers who buy products online in finding good deals and to save their money, efforts and their valuable time.

Keywords— Price Comparison, Web Scrapper, E-Commerce, Consumers, Product.

1.INTRODUCTION

Web Scrapper is the medium between the consumers and the sellers. In the modern era of e-commerce, consumers are faced with an overwhelming number of choices across various online platforms. This abundance often leads to difficulties in identifying the best deals for desired products. With the increasing reliance on online shopping, a need arises for application that streamline the process of comparing product prices across multiple e-commerce website.

Web scraper for product price analysis helps the customers to increase their price awareness by acing as tool to provide good deals. Unlike other comparison sites, E- commerce price comparison website (Web scrapper foe product price analysis) will focus on providing a price list of products which we want to search online and purchase at a cheaper price. By using web scraper for product price analysis, the customers don't feel misled by the various advertisements from the retailers who claim they offer the lowest price, but the reality is really different from what is shown to the customers. Due to the huge increase in online users, it will be of great help to those who have busy office work and don't have much time to check the current prices of products which they want to purchase. The study shows how connected people in India are to the Internet. In this the tools offers features that allow users to compare prices, identify the cheapest options, and sort products based on parameters such as price, customer reviews, and ratings. By automating the process of price comparison, this tool not only saves time but also empowers consumers to make cost-effective decisions. This paper introduces a web scraper-based price comparison tool designed to assist consumers in making informed purchasing decisions. By leveraging web scraping techniques using libraries like beautiful soup or Scrapy, the proposed tool extracts product details such as names, descriptions, prices, customer reviews, and ratings from multiple e-commerce platforms. The collected data undergoes cleaning and processing using pandas to ensure consistency and accuracy. In comparison to other countries, there are only few comparison sites are available in India. Most of them compare the price of hotel tariff, holiday package, mobile phone and others. As a consumer we all have the rights to choose which stores offers the best price for specific products. The price offered by any business, however, requires a lot of time and due to a limited time, you cannot compare prices and exit the purchase of certain products at a higher price. With a catalog that published online, sellers can save costs Increase the price awareness among consumers. This project has significant real-world applications, especially in a highly competitive market where price transparency can be a critical factor for consumer satisfaction.

2.LITERATURE REVIEW

The development of a web scraper-based price comparison tool draws on several domains, including web scraping, web mining, data processing, and e-commerce analytics. Literature survey is a critical component of any research study as it provides the overview of the existing literature on the chosen topic. A literature survey explores existing research and systems relevant to these domains, identifying gaps and providing a foundation for the proposed project.

2.1 Web Scrapping Techniques

Web scraping has become a widely used technique for extracting data from websites. Libraries like beautiful soup and Scrapy are popular tools for implementing scrapers, as they provide robust parsing capabilities and flexible data extraction mechanisms. Web scraping involves extracting data from websites, typically using automated tools or scripts. Web scraping is sending HTTP queries to web servers, receiving HTML material, and then parsing it to extract the needed data. It's similar to computerized copying and pasting data from a website into a document, but it's done programmatically and on a large scale.

2.2 Price Comparison Systems

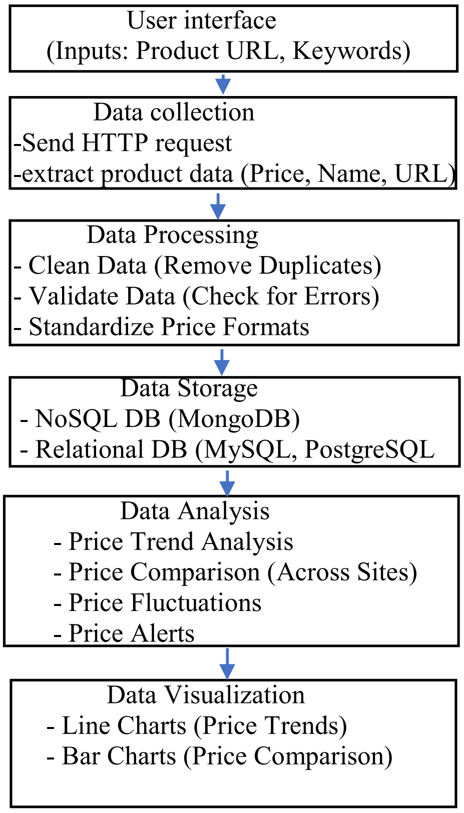
Existing price comparison systems, such as Google Shopping or PriceGrabber, demonstrate the utility of aggregating product data from multiple sources. These systems often rely on APIs provided by e-commerce platforms, which may limit their flexibility. This gap can be addressed by using web scraping to gather data directly from websites. The research centered on price comparison websites and the implications those websites have for the efficiency of markets and the level of price competition. Due to the fact that they have evolved into an aggregator of product information, price comparison websites are able to draw in all of the relevant stakeholders, including both customers and suppliers, to their own platforms.

2.3 Importance of Price Comparison

Price scrapping is used by the price comparison sites and E-Commerce sites of all types. It could be a simple Price Scraper using a chrome extension, a python script to scrape data from competing ecommerce websites like Walmart or Best Buy, or a full-fledged web scraping service. Retailers need to perform continuous price monitoring for several reasons. First, your competitors can sell the same product at a lower price or offer a more ludicrous discount. But Amazon managed to surpass this barrier by using web scraping and continued price monitoring. So let us understand Amazon's secret formula. Amazon leverages excellent price scraping to monitor competitors and offers products at competitive prices.

3.Proposed model

The proposed model aims to leverage web scraping techniques to collect product pricing data from multiple e-commerce platforms, followed by using deep learning models for accurate price prediction and trend analysis. The model consists of two major components: Data Collection and Price Prediction.


User interface
(Inputs: Product URL, Keywords)
Data collection
-Send HTTP request
-extract product data (Price, Name, URL)
Data Processing
- Clean Data (Remove Duplicates)
- Validate Data (Check for Errors)
- Standardize Price Formats
Data Storage
- NoSQL DB (MongoDB)
- Relational DB (MySQL, PostgreSQL
Data Analysis
- Price Trend Analysis
- Price Comparison (Across Sites)
- Price Fluctuations
- Price Alerts
Data Visualization
- Line Charts (Price Trends)
- Bar Charts (Price Comparison)

- Dashboards (Interactive Reports)

Fig.1 Proposed Model for Web Scraper for Product Price Analysis

3.1 User Interface

User Interface
The interface is simple and easy to use:
Inputs:
Users can either enter a product URL or search using keywords (e.g., "laptop," "headphones") to find products.
Product URL:
Simply paste the product link, and the system will fetch details like price and ratings.
Keywords:
Type in keywords to search multiple e-commerce sites for matching products.
This makes it easy for users to track prices, compare products, and set price alerts, all with minimal effort.

3.2 Data Collection

To develop a powerful and efficient model, the initial step involves gathering relevant data from e-commerce platforms using web scraping techniques. This data encompasses essential attributes like product names, prices, categories, seller details, ratings, and reviews.
Web Scraping Tools:
BeautifulSoup: This Python library allows parsing the HTML content of web pages, making it easy to extract the required product information.
Scrapy: Scrapy is used for more effective and scalable web crawling and scraping. It is especially useful when dealing with large volumes of data from multiple product pages that need to be scraped simultaneously.
Selenium: Selenium is employed for scraping data from websites that load content dynamically using JavaScript, which is a common occurrence on e-commerce sites.

3.3 Data Processing.

After collecting data from e-commerce websites, it’s important to clean, validate, and standardize it:
Clean Data (Remove Duplicates): We identify and remove duplicate entries, ensuring only unique products remain in the dataset.
Validate Data (Check for Errors): We check for missing values, unrealistic data (e.g., incorrect prices or ratings), and ensure consistency across all attributes. Errors are corrected or excluded.
Standardize Price Formats: Prices are standardized by converting to a single currency and ensuring consistent formatting (e.g., rounding to two decimal places). Discounts are handled by either keeping both original and discounted prices or selecting one.

3.4 Data Storage

For storing processed data, we have two options:
NoSQL DB (MongoDB):
Flexible and scalable, perfect for handling large, unstructured data.
Ideal for quick data retrieval and growth as the dataset expands.
Relational DB (MySQL, PostgreSQL):
Great for structured data with clear relationships (e.g., product info, prices).
Ensures data integrity and supports complex queries for detailed analysis.
Both options offer efficient storage, with MongoDB being more adaptable and MySQL/PostgreSQL better for structured, consistent data.

3.5 Data Analysis

Once the data is ready, we focus on key insights:
Price Trend Analysis:
We track how prices change over time to spot trends or patterns.
Price Comparison (Across Sites):
We compare prices for the same product across different websites to find the best deal.
Price Fluctuations:
We monitor how prices go up and down to understand pricing behavior.
Price Alerts:
Users can set alerts to get notified when a product hits their desired price.

3.6 Data Visualization

Data Visualization

We use simple visuals to make data easy to understand:
Line Charts: Show how prices change over time.
Bar Charts: Compare prices across different sites.
.

4.METHODOLOGY

Methodology part, we have highlighted the methodology of our proposed project. This section outlines the approach and techniques used to develop and evaluate a web scraper for product price analysis.

1. Define Objectives

Price Comparison: The first step in price analysis is to clearly define the objective of the analysis. For instance, is the goal to compare prices of a particular product across different e-commerce platforms or track historical price trends?
Scope of Data: Define the products or services to track and the websites from which to scrape data.

2. Identify Target Websites

Choose reliable and relevant websites that provide the required data.
Popular e-commerce websites (e.g., Amazon, eBay) or price comparison websites (e.g., PriceGrabber) are often scraped for this purpose.
Consider the frequency of updates (e.g., daily, weekly) based on the analysis requirements.

3. Legal and Ethical Considerations

Review Terms of Service (TOS): Ensure compliance with the websites' scraping policies and TOS. Some sites explicitly prohibit scraping.
Robots.txt: Check the website's robots.txt file to determine the allowed scraping permissions for different pages.
Rate Limiting: Avoid overloading websites with frequent requests by implementing rate limiting or using random intervals between scraping actions.

4. Data Extraction Methods

HTTP Requests: Use libraries like requests (Python) to send GET requests to the target web pages.
HTML Parsing: Once the page content is retrieved, parse the HTML to locate the relevant data using tools like: **Beautiful soup**(Python) for parsing and navigating HTML.
xml (Python) for fast XML and HTML parsing.
Selenium (for dynamic content): If the data is rendered by JavaScript, use Selenium or Puppeteer to interact with the page and extract data.

5. Data Cleaning

Extract the relevant data (e.g., price, product name, URL, etc.) by identifying the correct HTML elements or CSS selectors.
Clean and transform the data to remove irrelevant information, handle missing data, and standardize the format (e.g., removing currency symbols, converting text to numbers).

6. Price Analysis

Price Comparison: After scraping, compare prices across different websites or track the price changes over time. This can be done using basic statistics or more advanced models.
Trend Analysis: You can apply time series analysis or moving averages to identify pricing trends.
Price Volatility: Measure the fluctuations in prices over time using statistical methods like standard deviation or variance.
Discount Detection: Identify and analyses promotional discounts by comparing the sale price with the original price.

7. Data Storage and Visualization

Database: Store the extracted data in structured formats like CSV, JSON, or in a database (SQL, NoSQL) for further analysis.
Visualization: Use tools like Matplotlib, Seaborn (Python), or Excel for plotting graphs (e.g., price trends, distribution).

8. Automation and Monitoring

Scheduling Scrapes: Set up automated scraping schedules using tools like Cron jobs or task schedulers to scrape data at regular intervals.
Alerts: Set up price alerts to notify when a price drops below a certain threshold using email or SMS APIs.

9. Analysis and Reporting

Apply statistical techniques to Analise the price data, such as regression analysis, clustering, or machine learning models to predict future price trends.

Generate detailed reports to summarize insights, like the average price of a product across multiple platforms, price trends, or outliers in pricing.

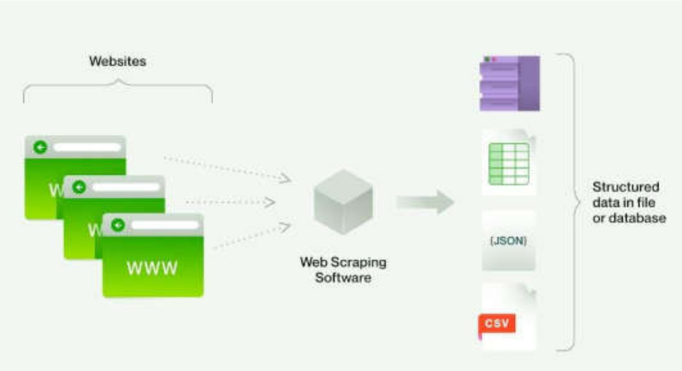


Fig.2 Web Scrapping.

5.Related Work

Web scraping is widely used for different projects, comparing prices online, observing changes in weather data, website change detection, research, integrating data from multiple sources, extracting offers and discounts, scraping job postings information from job portals, brand monitoring, and market analysis [16].

It is used for data collection. Web scraping has myriad applications in various domains. It acts as a prerequisite to big data analytics. Discussed below are a few of the several domains where web scraping is used.[17]

Social Media

Extracting data from social media platforms is a valuable tool for enhancing marketing campaigns. In today's fast-paced environment, businesses can quickly analyzes customer sentiment toward their products and improve public relations and audience engagement. To achieve this, researchers developed a web-based application for downloading Instagram account data, utilizing web scraping technology. This approach was chosen to avoid the limitations of Instagram’s Application Programming Interface (API), which restricts access to data on the platform. The web scraping method successfully created an Instagram data extractor application. The application was tested on 15 accounts, with the total number of posts ranging from 100 to 11,000. The solution successfully captured data from 2,412 Instagram accounts. The extracted data can be saved to a database and exported in multiple formats, including Excel, JSON, or CSV, providing users with flexibility in data management.[18].

Healthcare

Healthcare is no longer a field that depends solely on physical interactions. Instead, it has transitioned into the digital realm. In today’s data-driven world, web scraping in healthcare can play a crucial role in saving lives by enabling informed decision-making. Healthcare professionals often find the process of collecting data from numerous patients to be tedious and overwhelming. Despite the increasing need for clinical data, the current volume of patients makes it nearly impossible to gather it effectively. To address this, the author suggests the implementation of an automated system that can autonomously collect clinical data from SARS-CoV2 patients visiting the hospital for future research purposes. Additionally, web scraping techniques have been applied in healthcare research, as seen in the work of Dascalu et al. [19], where crawlers are used to extract drug leaflets.

Marketing

Boeger hausen et al. discuss the large amounts of customer data, often referred to as a digital footprint, which can be used to Analise customer behaviour and address research questions. In their paper, Saranya et al. propose using machine learning models to predict customer purchase intent during online transactions. They collect the necessary data through web scraping, as the information available on the web is often unstructured. This data is then Analise to forecast purchase intentions. Nguyen et al. examine the social media engagement of Australian SMEs by utilizing web scraping. They gather data from Instagram using the Instagram API and discover that tagging, rather than hashtags, leads to higher engagement, as tagging is perceived as more trustworthy.[17]

Finance

The author proposed an innovative approach to develop web-based indicators that could address the limitations of existing innovation metrics. Specifically, they created a strategy to identify product innovator companies on a large scale at minimal cost. Using traditional company-level data from a questionnaire-based innovation survey, they trained an Artificial Neural Network (ANN) classification model with label online texts of surveyed companies (from the German Community Innovation Survey). They then used this model to predict whether hundreds of thousands of German companies were product innovators by analyzing their online content. The predictions were compared against patent statistics, firm-level survey data, and regional innovation indicators. The results, which provide broad geographic coverage and granularity, demonstrate that this method offers reliable projections and could be a cost-effective and valuable addition to existing innovation indicators.

In a separate study, Tharanya et al. [20] conducted research using technical analysis on news articles scraped from the internet. They extracted news from a reputable website and summarized the content for further analysis and event modeling.

6.IMPLEMENTATION

v1. Project Planning - Stage 1

Initial stage of project planning, the problem related to the project is identified, and the significance of the study is assessed. The objectives and scope of the project are clearly defined, and the feasibility of completing the work within the given timeframe is evaluated. This stage involves studying the approach to the problem and determining the types of systems to be developed. Additionally, the tools and technologies to be used for the system's development are identified through a review of existing literature.

Possible Algorithm

S1: Clearly define the scope of the project.
S2: Identify the specific requirements for the project.
S3: Decompose the project into manageable tasks and estimate the required effort and resources.
S4: Develop a project schedule and allocate necessary resources.
S5: Conduct a risk assessment and implement mitigation strategies.
S6: Facilitate communication and collaboration, while continuously monitoring and controlling progress.
S7: Regularly review the project and make adjustments as needed to ensure successful completion.

Algorithm (Possible)

S1: Review and analyze system requirements,
S2: Identify key functionalities,
S3: Determine major components,
S4: Define interfaces and interactions,
S5: Define purpose and responsibilities,
S6: Input data, output data, and internal processing,
S7: Identify dependencies and interfaces,
S8: Define APIs and documentation,
S9: Verify and validate components against system requirements and design guidelines,
S10: Refine and iterate the drafting process, and document the drafted components.

V5. Develop System Architecture - Stage 5

The next section involves developing the structure of how the system will work. This will provide a clear understanding of the system's functionality and help avoid creating a system that does not address the problem it is intended to solve.

7. RESULT

After implementing the proposed model, we saw several positive outcomes:

Price Trend Analysis:

The model effectively tracked and displayed how prices changed over time, showing seasonal fluctuations and price drops, which helped identify the best times to buy.

Price Comparison:

We were able to easily compare prices of the same product across different e-commerce platforms, making it easier to find the best deals and save money.

Category	Product Name	Best Price	Platform
Electronics	One Plus 11r 5g	28,999	Amazon
Electronics	Hp Victus	74,499	Flipkart
Fashion	Red Tape white shoes	1710	Myntra
Skin Care	Cetaphil Moisturiser	649	Nykka

7.CONCLUSION

In conclusion, to locate the most optimal bargains web scraping serves as more than just a tool, it functions as a revolutionary force actively reshaping the e-commerce landscape. Users can access helpful information on the website, which will assist them in making decisions that are in their best interests. By equipping consumers with actionable insights and revealing the concealed mechanisms of online marketplaces, it establishes the groundwork for a shopping experience that is not only more knowledgeable but also more transparent. It is now possible for working people to check on the price of things before making purchases, as a result of the existence of a website that compares prices. Users of this website will be able to compare costs on a variety of e-commerce shopping websites in order to choose which website offers the best combination of low cost and a good deal on the product they are interested in purchasing. Our project stands as a testament to the potential for transformation that web scraping possesses, as it effectively demonstrates how this technique can provide individuals with the means to make more intelligent purchasing decisions and enable them to navigate the digital marketplace with astuteness.

REFERENCES

[1] M. Brahimi, K. Boukhalfa, and F. Moussaoui, "Deep learning for tomato diseases: Classification and symptoms visualization," *Applied Artificial Intelligence*, vol. 31, no. 4, pp. 299–315, 2017

[2] S. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in Plant Science*, vol. 7, p. 1419, 2016.

[3] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic,"Deep neural networks based recognition of plant diseases by leaf image classification," *Computational Intelligence and Neuroscience*, vol. 2016, pp. 1–11, 2016.

[4] J. Amara, B. Bouaziz, and A. Algergawy, "A deep learning-based approach for banana leaf diseases classification," in *Proceedings of the IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, 2017, pp. 1–7.

[5] A. Brahimi, S. Boukhalfa, and A. Moussaoui, "Tomato plant diseases detection using deep learning," in *Proceedings of the IEEE International Conference on Artificial Intelligence and Computer Vision (AICV)*, 2020, pp. 126–132.

[6] A. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, 2018.

[7] R. Ramcharan, A. Baranowski, P. McCloskey, K. Ahmed, S. Legg, and J. Hughes, "Deep learning for image-based cassava disease detection," *Frontiers in Plant Science*, vol. 8, p. 1852, 2017.

[8] L. Sun, W. Jiang, and H. Zhang, "Plant disease identification based on deep learning algorithm in smart farming," *Discrete Dynamics in Nature and Society*, vol. 2018, Article ID 2479172, 2018.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPS)*, 2012, pp. 1097–1105.

[10] S. P. Singh, D. N. Bharathi, and M. S. Rao, "Plant disease identification using CNN and SVM techniques: A review," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 9, no. 1, pp. 2294–2301, 2019.

[11] P. Mohan, K. D. Naveen, and K. Manogaran, "Disease detection on plant leaf using image segmentation and soft computing techniques," *Journal of Artificial Intelligence*, vol. 12, pp. 104–118, 2019.

[12] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.

[13] "Web Scraping with Python: Collecting Data from the Modern Web" by Ryan Mitchell(O'Reilly Media).

[14]Documentation from Python libraries such as BeautifulSoup, Scrapy, and Pandas.

[15] Online tutorials and case studies on data extraction for price comparison and trend analysis.

[16] Sirisuriya, D. S. (2015). A comparative study on web scraping. In the Proc. 8th Int. Res. Conf. KDU, 135– 140.

[17] Chaimaa Lotfi, Swetha Srinivasan, Myriam Ertz, Imen Latrous LaboNFC, University of Quebec at Chicoutimi, 555 Boulevard de l'Université, Saguenay (QC), Canada

[18] SCRS Conference Proceedings on Intelligent Systems (2021), page 383.

[19] Phan, H. (2019). Building Application Powered by Web Scraping. Doctoral Thesis.

[20] Tharaniya, B. et al. (2018). Extracting Unstructured Data and Analysis and Prediction of Financial Event Modeling. In Conference proceedings of the Annual Conference IET, 6-11.

[21] Suganya, E. and Vijayarani, S. (2021). Firefly Optimization Algorithm Based Web Scraping for Web Citation Extraction. Wireless Personal Communications, 118(2):1481-1505.

[22] Rahmatulloh, A. and Gunawan, R. (2020). Web Scraping with HTML DOM Method for Data Collection of Scientific Articles from Google Scholar. Indonesian Journal of Information Systems, 2(2):95-104.

[23] Kolli, S., Krishna, P. R. and Reddy, P. B. (2006). A novel NLP and Machine Learning based text extraction approach from online news feed. ARPN Journal of Engineering and Applied Sciences, 16(6):679-685.

[24] Li, R. Y. M. (2020). Building updated research agenda by investigating papers indexed on Google scholar: A natural language processing approach. In International Conference on Applied Human Factors and Ergonomics. Springer, Cham, 298-305.

[25] Ertz, M. (2022). Handbook of research on the platform economy and the evolution of e-commerce. Hershey, PA: IGI Global.