Archives available at journals.mriindia.com

International Journal of Advanced Scientific Research and Engineering Trends

ISSN: 2456-0774

Volume 10 Issue 01, 2026

Retail Purchase Intelligence System

[1]Sakshi Shivaji Godse, [2]Darshan Yogesh Kangane, [3]Vyankatesh Gopaldas Bairagi, [4]Vishakha Rajendra Ganore, [5]Abhay Gaidhani

[1,2,3,4,5] Computer Department, Sandip Institute of technology and research centre, Nashik, India

[1]sakshi.godse19@gmail.com,[2] kanganedarshan06@gmail.com, [3]Vyankateshbairagi.dev@gmail.com, [4]vishakhaganore14@gmail.com, [5]abhay.gaidhani@sitrc.org

Peer Review Information

Submission: 21 Dec 2025
Revision: 05 Jan 2026
Acceptance: 18 Jan 2026

Keywords

E-commerce, Web Scraping, Price Comparison, Python, Data Aggregation, Consumer Intelligence, Online Retail, Automation

Abstract

Context and Motivation: The growing popularity of e-commerce platforms has transformed consumer behavior, with modern buyers increasingly relying on digital channels to compare prices before making purchases. However, manual price checking across multiple websites remains inefficient, time-consuming, and error-prone. This research presents a Retail Purchase Intelligence System, an automated price comparison framework that aggregates product pricing information from various e-commerce sources and displays it in a unified interface. The system utilizes web scraping techniques through Python libraries such as Beautiful Soup and Requests, combined with a centralized MySQL database for structured data storage. A lightweight front-end interface built with HTML, CSS, and JavaScript enables intuitive search and quick visualization of comparative results. Experimental validation demonstrates that the system can accurately extract and normalize pricing data across multiple online retailers, significantly reducing consumer effort and time in finding optimal deals. The proposed model also outlines scalability for dynamic websites through Selenium-based scraping and highlights future extensions such as price-trend analysis, alert notifications, and browser integration. Overall, the system provides an effective, low-cost solution for real-time price intelligence and contributes to advancing consumer-centric automation in digital retail.

Introduction

The explosive growth of e-commerce has revolutionized the way consumers interact with products and sellers worldwide. According to global retail reports, more than 80% of online buyers compare prices across multiple stores before finalizing a purchase. This behavioral pattern highlights a key consumer priority—price optimization—which continues to shape competition among online retailers. In traditional practice, buyers manually browse through numerous e-commerce platforms such as Amazon, Flipkart, or Snapdeal to verify product prices, reviews, and delivery options. While this method ensures user control, it is inherently time- consuming, prone to human error, and lacks real-time visibility of market fluctuations. The need for an automated solution that streamlines this process has become evident as online inventories and product variations increase rapidly. Recent studies on digital purchasing behavior reveal that consumers spend up to 30% of their decision time comparing prices rather than evaluating quality or reviews [1], [2]. Platforms like Google Shopping and Honey partially address this gap, yet they often face limitations when it comes to region- specific retailers, product normalization,

and scraping dynamically rendered pages. The proposed work—Retail Purchase Intelligence System— aims to automate and enhance this process through a scalable, lightweight, and easily deployable solution. The system employs web scraping and HTML parsing techniques to extract structured data such as product names, pricing, and URLs from multiple sources. Extracted data are stored in a centralized MySQL database and presented to users via a responsiveweb-based interface built with HTML, CSS, and JavaScript.

Literature Survey

The rapid growth of e-commerce ecosystems has led to an increasing demand for intelligent systems capable of assisting consumers in making informed purchasing decisions. Price comparison platforms have emerged as an essential component of digital retail, enabling users to evaluate multiple sellers simultaneously. Several researchers have explored automated price aggregation, web scraping methodologies, and intelligent decision-support systems to address challenges associated with online product comparison.

Early studies in e-commerce price comparison focused on network-based models that analyzed interactions between retailers and comparison platforms. Beranek and Remes proposed an e-commerce network framework emphasizing synchronization challenges among distributed vendor systems, highlighting the complexity of maintaining accurate and up-to-date pricing information across multiple marketplaces. Their work demonstrated that real-time aggregation requires efficient data acquisition mechanisms and scalable architectures capable of handling heterogeneous data sources.

With the advancement of data analytics, researchers began examining pricing strategies influenced by big data technologies. Wen investigated pricing discrimination strategies adopted by online sellers using consumer behavioral analytics, revealing how dynamic pricing mechanisms continuously modify product costs based on demand patterns, user activity, and competitive market conditions. This research emphasized the importance of automated monitoring systems capable of tracking price fluctuations in real time to support transparent consumer decision-making.

Supply-chain-oriented studies further explored pricing coordination between manufacturers and retailers. Bao et al. analyzed service-level agreements and pricing relationships within manufacturer–retailer ecosystems, demonstrating that variations in logistics efficiency and service quality significantly influence final product pricing. These findings reinforced the need for consumer-facing systems that aggregate pricing data from multiple vendors to provide comprehensive market visibility.

Recent research has increasingly focused on implementing practical price comparison platforms using web scraping technologies. Shaikh and Patel developed a Python-based price comparison website utilizing Beautiful Soup for extracting product data from static HTML pages. Their study validated the feasibility of lightweight scraping tools for structured data extraction but also identified limitations when dealing with dynamically rendered web content. Similarly, Alam et al. introduced UPOMA, a localized price comparison system designed for regional e-commerce platforms, integrating live scraping with responsive web interfaces. Their work demonstrated the effectiveness of combining backend automation with user-friendly visualization for enhancing consumer experience.

A major challenge identified across studies is product matching and entity normalization. Different e-commerce websites often present identical products using varying naming conventions, formats, and attribute descriptions, leading to inconsistencies during aggregation. Li et al. proposed deep entity-matching approaches using pre-trained language models to improve cross-platform product identification accuracy. Likewise, Tóth et al. introduced multimodal matching techniques that combine textual and visual product features to enhance comparison accuracy, particularly in fashion e-commerce environments. These approaches highlight the growing role of artificial intelligence in improving data alignment across heterogeneous platforms.

Researchers have also compared multiple web scraping frameworks to determine efficient extraction strategies. Choudhary and Thakur demonstrated that hybrid scraping techniques combining static parsing and dynamic browser automation significantly improve extraction success rates on JavaScript-heavy websites. Patil et al. further supported this conclusion by showing that Selenium-based automation enables reliable data acquisition from modern interactive web applications where traditional HTTP-based scraping fails.

Beyond price comparison, several studies explored enhancing consumer intelligence through additional analytical features. Harshitha et al. and Harini et al. emphasized integrating review analytics, price alerts, and recommendation mechanisms into comparison platforms to provide deeper insights beyond

basic price listings. These enhancements transform comparison tools into intelligent decision-support systems capable of predicting trends and assisting users proactively.

Despite significant advancements, existing solutions still face several limitations. Many commercial platforms restrict coverage to selected retailers, lack transparency in data sourcing, or struggle with frequent structural changes in website layouts. Ethical concerns related to scraping policies, rate limiting, and scalability also remain active research challenges. Furthermore, most academic implementations primarily focus on data extraction rather than incorporating analytical intelligence such as price trend evaluation or predictive modeling.

The Retail Purchase Intelligence System builds upon these prior contributions by integrating automated web scraping, structured data storage, and unified visualization within a scalable architecture. Unlike earlier systems that rely solely on static comparison, the proposed enhanced framework aims to extend functionality toward intelligent retail analytics, enabling efficient data normalization, scalable scraping for dynamic websites, and future integration of predictive and alert-based modules. By addressing limitations identified in existing literature, the system contributes toward developing a comprehensive consumer-centric price intelligence solution for modern digital marketplaces.

Problem Statement

The exponential growth of online retail platforms has fundamentally reshaped the purchasing behavior of consumers, enabling access to a vast range of products across multiple e-commerce marketplaces. While this digital transformation offers convenience and variety, it also introduces significant challenges for consumers attempting to identify the most cost-effective purchasing options. Buyers frequently visit multiple websites to compare prices, delivery options, and product availability before making a final decision. This manual comparison process is inefficient, repetitive, and time-consuming, often requiring users to navigate numerous browser tabs and interpret inconsistent product descriptions across platforms.

A major difficulty arises from the lack of standardized product representation among e-commerce websites. Identical products are frequently listed under different names, formats, or attribute structures, making direct comparison challenging and increasing the likelihood of inaccurate decision-making. Furthermore, rapid price fluctuations driven by competitive pricing strategies, promotional campaigns, and algorithmic pricing models make it difficult for consumers to monitor real-time changes effectively.

Existing commercial price comparison tools provide partial automation; however, they often suffer from limited retailer coverage, restricted regional adaptability, and inadequate transparency in data collection mechanisms. Many platforms also struggle to extract information from dynamically rendered websites that rely heavily on JavaScript, resulting in incomplete or outdated pricing data. Additionally, most available systems focus solely on displaying current prices without offering analytical insights such as historical trends, intelligent ranking, or predictive assistance.

From a technical perspective, frequent structural modifications in website layouts, anti-scraping mechanisms, and inconsistent data formats present challenges in building scalable and reliable comparison systems. These limitations highlight the need for a robust, automated framework capable of extracting, normalizing, and aggregating product information from diverse sources while ensuring accuracy, scalability, and usability.

Therefore, the core problem addressed in this research is the design and development of an intelligent Retail Purchase Intelligence System that automates multi-platform price aggregation, standardizes heterogeneous product data, and presents unified comparative insights through an interactive web interface. The system aims to minimize consumer effort, improve decision efficiency, and establish a scalable foundation for future enhancements such as price trend analysis, alert mechanisms, and intelligent retail analytics.

Objectives

The primary objective of this research is to design and implement an advanced Retail Purchase Intelligence System that automates the process of online product price comparison while enhancing consumer decision-making through efficient data aggregation and visualization. To achieve this goal, the study defines the following specific objectives:

1. **To develop an automated data extraction framework** Design a reliable web scraping mechanism using Python-based technologies capable of collecting product-related information such as titles, prices, and purchase links from multiple e-commerce platforms. The framework should support both static and dynamically rendered web pages to ensure comprehensive data acquisition.

2. To implement a centralized data management system Establish a structured MySQL database to store extracted information efficiently, enabling organized data retrieval, faster querying, and maintainable storage of normalized product records for analysis and comparison.

3. To standardize and normalize heterogeneous product data Develop preprocessing and normalization techniques that align product attributes obtained from different websites, reducing duplication and improving comparison accuracy through consistent data representation.

4. To design an intuitive and responsive user interface Create a lightweight web-based interface using HTML, CSS, and JavaScript that allows users to perform product searches easily and visualize comparative pricing results in a clear and interactive format.

5. To enhance user navigation and purchasing efficiency Integrate direct redirection mechanisms that enable users to access retailer websites instantly, reducing search effort and improving overall usability.

6. To evaluate system performance and accuracy Analyze the efficiency of the scraping engine, response time, and correctness of aggregated data through experimental validation and comparative testing across multiple e-commerce sources.

7. To ensure scalability and extensibility of the framework Design the system architecture in a modular manner to support future enhancements such as price history tracking, automated price-drop alerts, predictive analytics, and browser extension integration.

Collectively, these objectives aim to deliver a scalable, automated, and consumer-centric solution that bridges the gap between raw e-commerce data and intelligent purchasing support systems.

System Architecture

Overview

The Retail Purchase Intelligence System adopts a modular client–server architecture designed to transform a user's product query into a comprehensive and unified price comparison output. The architecture integrates automated data acquisition,

processing, storage, and visualization components to ensure efficient system performance and scalability. The overall workflow begins with user interaction through a web interface and progresses through multiple processing layers before presenting normalized comparative results.

The architecture consists of interconnected modules including the Web User Interface, Request Handling Layer, Scraper Engine, Data Normalization and Aggregation Module, Database Management System, Administrative Control Panel, and External E-commerce Data Sources.

1. Web User Interface (Frontend Layer)

The frontend module serves as the interaction point between users and the system. Developed using HTML, CSS, and JavaScript, the interface allows users to enter product search queries and visualize price comparisons in a structured tabular format. The interface emphasizes usability, responsiveness, and clarity, ensuring that users can quickly interpret pricing differences across retailers. Hyperlinks to original product pages are provided to enable seamless purchase navigation.

2. Request Handler and API Layer

The request handling component acts as an intermediary between the frontend and backend services. It validates user inputs, processes search requests, and manages communication with the scraping engine. This layer also implements rate limiting and task scheduling mechanisms to maintain system stability and prevent excessive requests to external websites.

3. Scraper Engine (Data Acquisition Layer)

The scraper engine is responsible for extracting product information from multiple e-commerce platforms using Python-based technologies.

- **Static Content Extraction**: Utilizes HTTP requests and HTML parsing techniques to retrieve product titles, prices, and URLs from standard web pages.
- **Dynamic Content Extraction**: Employs Selenium-based browser automation to handle JavaScript-rendered pages where traditional scraping techniques are ineffective.
- **Resilience Mechanisms**: Includes retry strategies, selector versioning, and controlled request delays to ensure reliable operation despite frequent website structural changes.

This hybrid approach enhances extraction accuracy and expands compatibility with modern web architectures.

4. Data Normalizer and Aggregator

Extracted data from different sources often vary in format and structure. The normalization module cleans and standardizes product attributes such as currency formats, naming conventions, and textual variations. Lightweight

matching rules based on product keywords, brand identifiers, and attributes are applied to merge duplicate entries. The aggregation process ranks results according to pricing and data freshness, improving comparison quality.

5. MySQL Data Store

The centralized database stores both raw scraped data and processed records. It supports efficient querying, incremental updates, and historical tracking capabilities. The database structure ensures data consistency while enabling future analytical extensions such as price trend monitoring.

6. Admin Control Panel

An administrative interface allows system maintainers to manage website configurations, update scraping selectors, monitor logs, and adjust operational parameters. This module ensures adaptability when external websites modify their layouts or data structures.

7. External E-commerce Sources

Online retail platforms act as authoritative data providers. The system accesses these platforms in a read-only manner to retrieve publicly available product information required for comparison. Ethical scraping practices, including controlled request frequency and compliance considerations, are incorporated to maintain responsible data acquisition.

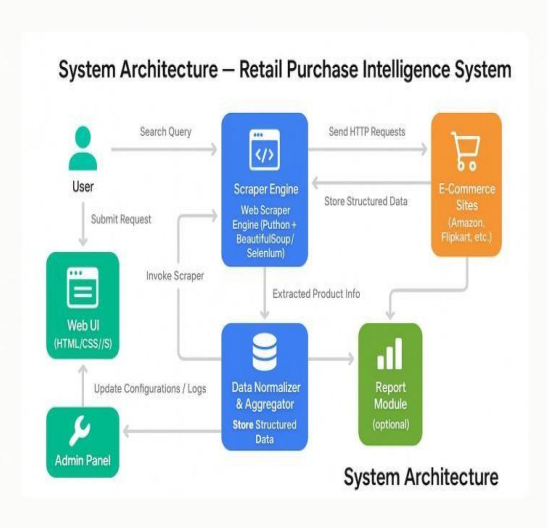

Figure 1: Illustrates the high-level architecture used in implementation

References

L. Beranek and R. Remes, "E-commerce network with price comparator sites," in Proc. 9th Int. Conf. on Advanced Computer Information Technologies (ACIT), 2019, pp. 401–404. doi: 10.1109/ACITT.2019.8779865.

P. Wen, "Seller's pricing discrimination strategies under adoption of online big data technology," in Proc. Int. Conf. on ECommerce and E-Management (ICECEM), 2021, pp. 201–206. doi: 10.1109/ICECEM54757.2021.00047.

F. Bao, D. Chen, H. Miao, and Y. Chen, "TPL service level and pricing models in manufacturer–retailer supply chains," in Proc. Int. Conf. on Management of E-Commerce and E-Government (ICMeCG), 2010, pp. 333–337. doi: 10.1109/ICMeCG.2010.75. ijirmps.org

S. M. Shaikh and A. Patel, "E-commerce price comparison website using web scraping," Int. J. Innovative Research in Multidisciplinary and Professional Studies (IJIRMPS), vol. 11, no. 3, pp. 45–49, 2023. [Online]. Available: https://www.ijirmps.org/papers/2023/3/230223.pdf. doi: 10.37082/IJIRMPS.v11.i3.230223.

A. Alam, A. A. Anjum, F. S. Tasin, M. R. Reyad, S. A. Sinthee, and N. Hossain, "Upoma: A dynamic online price comparison tool for Bangladeshi e-commerce websites," in Proc. IEEE Region 10 Symposium (TENSYMP), 2020, pp. 194–197. doi: 10.1109/TENSYMP50017.2020.9230862. [PDF].

N. Patil et al., "Product price comparison across several ecommerce websites," IJIRSET, vol. 13, no. 5, pp. 1759–1765, 2024. [PDF].

P. Rathod, M. Jadhav, and P. Pawar, "E-commerce with price comparison, price alert and fake review detection," IJARCCE, vol. 10, no. 1, pp. 67–72, 2021. [PDF].

A. Choudhary and S. Thakur, "Web scraping-based product comparison model for e-commerce," JETIR, vol. 11, no. 4, pp. 238–244, 2024. [PDF].

A. Kumar and S. Gupta, "Product comparison in ecommerce," IRJET, vol. 11, no. 3, pp. 1140–1146, 2024. [PDF].

V. B. Raj et al., "Price Probe: E-commerce platforms using machine learning," IJFMR, vol. 6, no. 3, pp. 1–9, 2024. [PDF]. IJFMR

W3C, "WebDriver (Level 2)," W3C Candidate Recommendation, Sept. 2025. [Online]. Available:

Selenium Project, "WebDriver documentation," Nov. 2024. [Online]. Available: https://www.selenium.dev/documentation/webdriver/ Selenium

L. Richardson, "Beautiful Soup 4.13

documentation," 2025. [Online]. Available: https://www.crummy.com/software/BeautifulSoup/bs4/doc/ crummy.com

Y. Li, J. Li, Y. Suhara, A. Doan, and W.-C. Tan, "Deep entity matching with pre-trained language models," Proc. VLDB Endowment (PVLDB), vol. 14, no. 1, pp. 50–60, 2021. doi: 10.14778/3421424.3421431. [PDF].

S. Tóth et al., "End-to-end multi-modal product matching in fashion e-commerce," arXiv preprint arXiv:2403.11593, 2024. [Online]. Available: https://arxiv.org/abs/2403.11593 arXiv