

SmartShopper: An AI-Based Price Tracking and Prediction System for Online Deals

Jadhav Shreyash | Janga Vignesh

Department of Computer Science and Engineering (Data Science)

Mahatma Gandhi Institute of Technology (Autonomous), Hyderabad, Telangana - 500075, India Guided By:

Dr. Shaik Irfan Babu, Assistant Professor, Department of Emerging Technologies [Emails :](mailto:irfanbabu@mgit.ac.in)

irfanbabu@mgit.ac.in

jshreyash_csd226726@mgit.ac.in, jvignesh_csd226727@mgit.ac.in

ABSTRACT

Online prices tend to be volatile due to flash sales, secret discounts, and other factors making it difficult to identify when one should purchase products. Finding an optimal time appears impossible as the best deals keep changing. Fortunately, there is SmartShopper. SmartShopper is an artificial intelligence-powered tool that monitors prices, analyzes price fluctuations using machine learning methods, and helps consumers know if they are making a worthwhile purchase. It employs random forest regression algorithms for short-time price spikes and LSTM models for long-term analysis. Web scrapers feed these systems with updated price information continuously. For long-term predictions, the system uses Google Gemini AI, predicting future price trends two years ahead. SmartShopper performs quickly, fetching the price information within one second while alerting consumers about the optimal time to purchase items, be it a flash sale or Black Friday. The tool saves customers both time and money.

Keywords: e-commerce, price tracking, machine learning, LSTM, random forest, web scraping, price forecasting, AI, Google Gemini

I. INTRODUCTION

There has been a boom in online shopping. But with increasing number of users in online shopping sites, their prices become extremely unpredictable there can be special sales, discounts that come unexpectedly, dynamic prices determined by algorithms, and extreme competition, which makes it quite impossible for the users to judge if it is a good time to buy or not. Yes, there will be some sort of basic system on the site itself to show prices through basic price graphs or something similar. But those are not only ineffective but do not capture those times when the prices are at rock bottom.

A. Motivation

At present, shoppers jump back and forth from outdated methods of finding sales, unplanned sale alerts, or even rudimentary plug-ins that fail to cope during fast-changing price periods. Such primitive methods result in lost opportunities and excessive spending on products. However, thanks to artificial intelligence being capable of handling vast amounts of product information, smarter shopping is achievable at last.

B. Problem Statement

Prices on the Internet are highly volatile, which means that consumers pay for products beyond their actual value. The solutions offered currently, which include manual checking or simple plug-ins, do not guarantee accuracy and compatibility across all devices. SmartShopper solves this problem through consistent monitoring of prices, predictions, and alerts regarding when to buy products.

II. RELATED WORK

Price monitoring has attracted much attention in the field of e-commerce. A literature review of ten most important studies related to price prediction with regards to classical machine learning, deep learning, and web scraping approaches was carried out. Main findings include the following: Nonlinear and nonconforming nature of prices is better forecasted through Random Forest and LSTM; Available price prediction methods deal with no more than one online store at once; and Web scraping and price prediction are rarely combined..

Table I: Literature Survey

#	Author(s)	Year	Methodology	Merits	Limitations
1	Ferreira et al.	2016	ML-based demand & dynamic price optimization for flash-sale markets.	Real-world deployment with revenue gains.	Limited to specific market types.
2	Enache	2021	Survey of ML techniques for price elasticity and competitive pricing.	Broad coverage of pricing strategies.	Lacks implementation depth.
3	Kaur	2023	Comparative study of ML models for product price prediction.	Clear performance comparison; interpretable.	Small dataset; missing advanced models.
4	Dhulipalla et al.	2025	Web scraping pipeline for real-time multi-platform price collection.	Accurate multi-site tracking.	No forecasting; brittle scrapers.
5	Song	2022	Rolling-horizon forecasting via ARIMA, SVM, RF.	Accurate short-term forecasts.	Not directly price-focused.
6	Pathe	2022	Regression & ensemble models for automated price suggestion.	End-to-end practical implementation.	Limited dataset categories.
7	Zhang & Wang	2020	Q-learning reinforcement learning for competitive pricing.	Strong adaptability in markets.	Requires large training datasets.
8	Bukhari & Ahmad	2021	LSTM-based sequence forecasting for nonlinear price patterns.	Captures complex sequential patterns.	High training cost.
9	Li & Fei	2019	Big-data ML pipeline for demand and pricing trend prediction.	Scalable for large datasets.	Expensive infrastructure.
10	Chen & Liu	2022	Gradient Boosting with competitor price fusion for forecasting.	High accuracy multi-seller.	Low interpretability.

Observations based on the survey:

- ML-based methods (Random Forest, LSTM) work better than statistical techniques for forecasting non-linear time-series data;
- Very few systems provide comparative analysis across different platforms;
- Few studies combine real-time data collection and forecasting in one web app. SmartShopper resolves these issues.

III. SYSTEM ARCHITECTURE AND METHODOLOGY

SmartShop brings all of that together in four components: Input, Processing, Database, and Output. These four components have their own functions, but they work as a complete solution for seamless scaling..

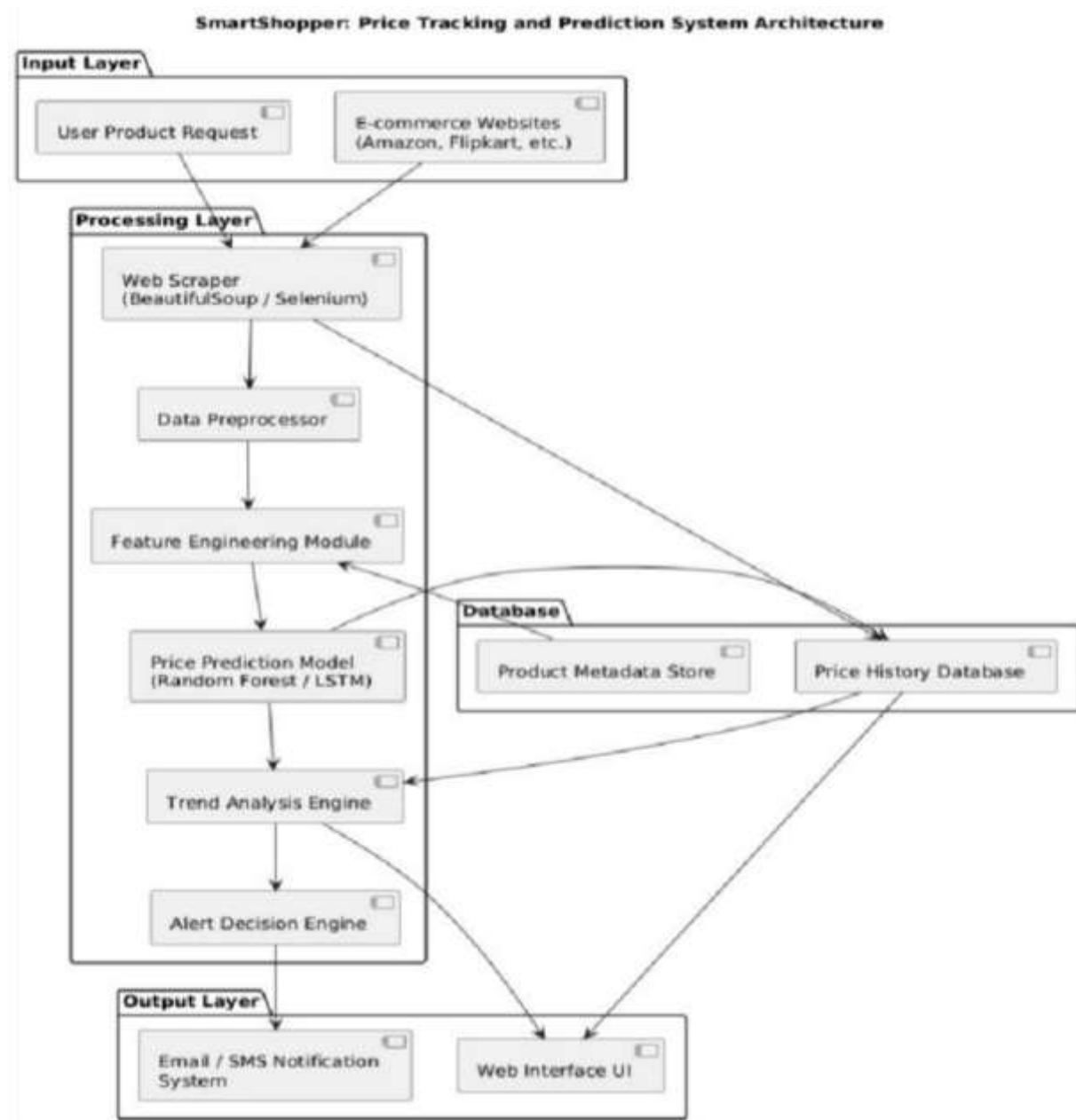


Fig. 1. SmartShopper System Architecture — Four-Layer Design

A. System Architecture Overview

Input Layer: This layer collects the user's preferences – just enter the product that you want to buy or the link that you want to reach, and the algorithm will take you straight to those websites. **The Processing Layer:** Here all the processing happens – extracting the product page, cleaning and processing the raw data, creating new features, predicting, performing time-series analysis, and finally triggering alerts for any relevant updates. All this data is safely stored in an SQLite3 database. Outputs can be through the UI or emails using Resend API or even SMS/Push Notifications..

B. UML Diagrams

The following UML diagrams illustrate the static structure and dynamic behavior of the SmartShopper system.

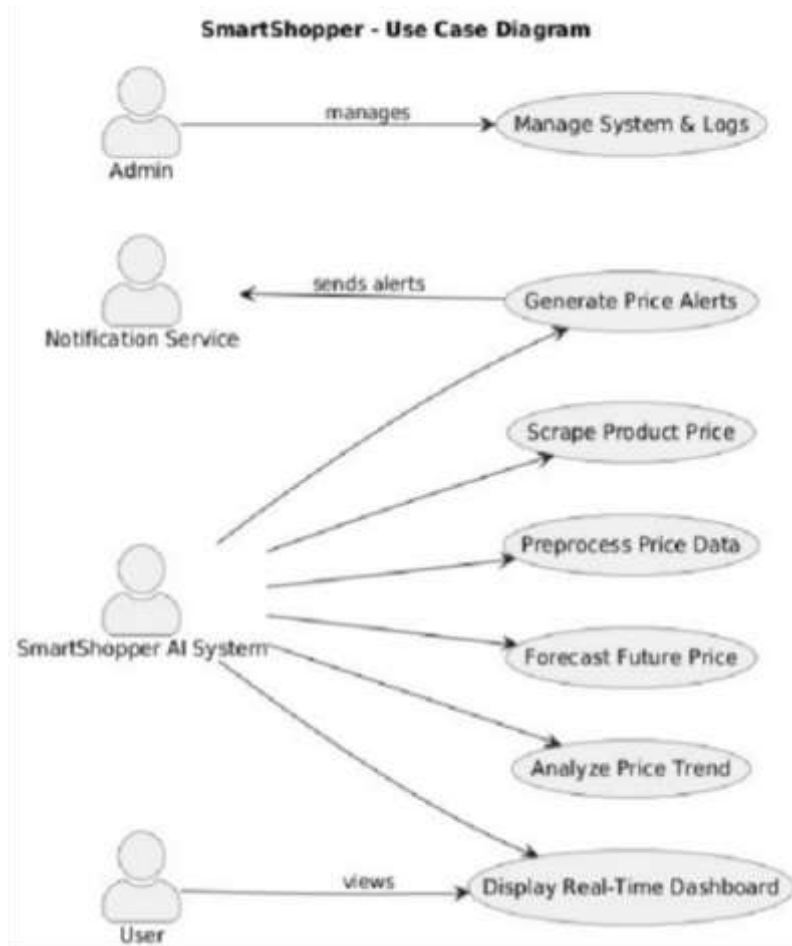


Fig. 2. Use Case Diagram — SmartShopper System Actors and Actions

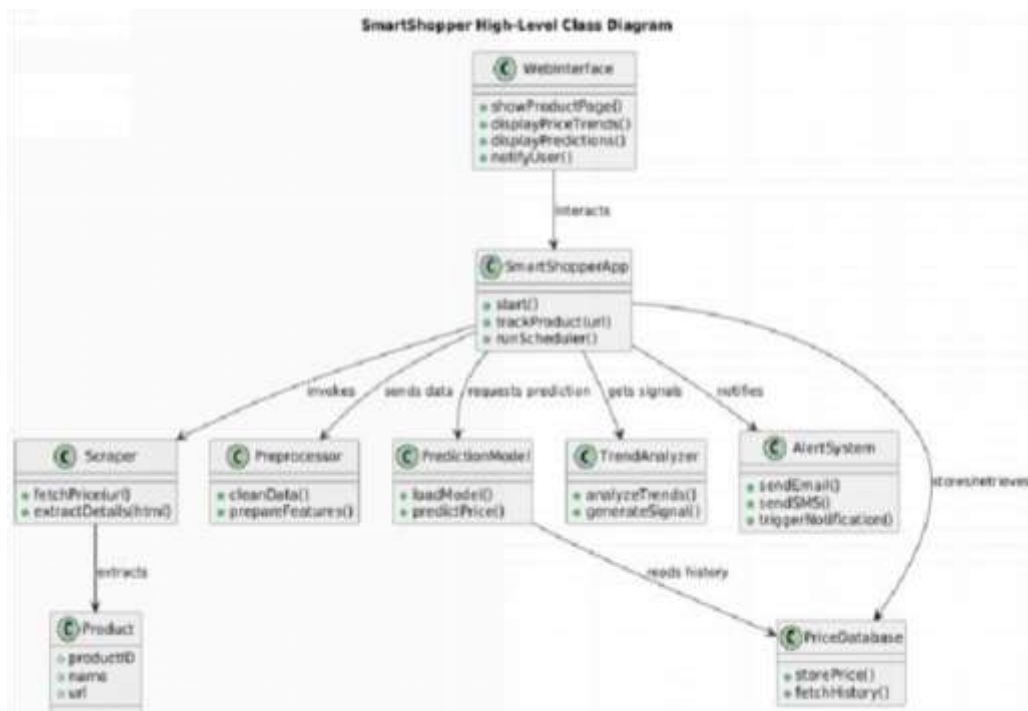


Fig. 3. Class Diagram — SmartShopper Module Interactions

C. Data Collection and Preprocessing

Data on price is collected via the use of the live website such as Amazon and Flipkart through the use of web scraping methods that we have implemented ourselves via the utilization of libraries such as BeautifulSoup, Cheerio, and Selenium. It is done on all types of products including electronic gadgets, home appliances, and even beauty products.

D. Prediction Models

The SmartShopper algorithm utilizes two principal machine learning models:

- Forest Regression: effective for detecting nonlinear and short-range price movements. It combines decision trees based on current price data, seasonality, and trend indicators. Mean squared error is used as the loss function, while Adam is the optimizer.
- LSTM Neural Networks: ideal for capturing long-range trends in which past occurrences influence future outcomes. Performance is evaluated using RMSE, MAPE, and accuracy metrics. The most accurate models are stored for quick future deployment.

E. Google Gemini AI Integration

We use highly specific prompts to query Google Gemini (gemini-1.5-flash) to obtain comprehensive historical price trends for both a full year behind and a full year ahead of time, all in convenient JSON format. Should Gemini experience downtime, we fall back to other models, or even a simple price prediction. For instance, in case of a Samsung Galaxy S24, which costs Rs. 49,999, Gemini predicted substantial increases in Q4 due to sales and milder pricing in Q1.

F. Alert and Notification Mechanism

SmartShopper triggers notifications for both large drops or positive trends according to user-defined thresholds. E-mail notifications are instantly sent through the Resend API, and additionally, you have an option to receive push notifications or SMS. Notification information includes details about the product, its price, predicted price drop, date and time stamp, and buying link, but you won't have to worry about spam notifications..

IV. IMPLEMENTATION

Node.js is our language on the backend along with Express framework while the front-end is developed using plain JavaScript for creating a one-page application. Our authentication service is provided using Google OAuth, Firebase while SQLite3 is used to manage our database. Our backend has a scheduled task that operates every hour by scraping up to 50 items per time (two in parallel to avoid IP blocking). OLS regression starts when there is a minimum of three points for price forecasting..

A. Tech Stack Summary

Component	Technology / Detail
Operating System	Windows 11 (64-bit)
Runtime	Node.js v20.x (LTS)
Server Framework	Express.js v5.2.1
Database	SQLite3 v6.0.1
Frontend	Vanilla HTML5 / CSS3 / JavaScript (SPA)
API Integrations	SerpApi, Apify, ScraperAPI, Google Gemini AI, Resend
Authentication	Firebase Authentication (Google OAuth 2.0)

V. RESULTS AND DISCUSSION

A. Keyword-Based Product Search

Tests conducted on common search terms such as 'boAt Rockerz 255 Pro', 'OnePlus Nord Buds 2', and 'Himalaya face wash men' using Google India (INR) revealed that the search engine was quite effective as it ignored the use of stop words and used keyword matching criteria, requiring at least 40% matches to display results.



Fig. 4. SmartShopper Search Interface — Keyword-Based Product Query

B. In-Memory Caching Performance

Our in-memory cache made repeat searches almost instant. What took 1,800 ms the first time dropped to under 10 ms on repeats.

Table II: Cache Performance

Request	Response Time	Cache Status
First Request	~1,800 ms	CACHE MISS
Second Request	< 10 ms	CACHE HIT

The in-memory cache reduced response times from approximately 1,800 ms to under 10 ms — a 180× improvement — for repeated queries within the cache expiration window.

E. AI Price Forecasting – Google Gemini & Linear Regression Model

The AI prediction using Google Gemini produces 2-year price lifecycles for each product. In Fig. 5 below is the two-year market context where historical data was used to produce forecasted prices. Given the history of India's e-commerce market and the seasonal variations (with Diwali spikes during Q4 and drops during Q1), the AI was able to generate seasonal price fluctuations. Fig. 6 is the price forecast model with historical data beginning from April 2025 up to April 2027.



Fig. 5. 2-Year Market Context Panel — AI-Generated Historical Forecasting Dataset



Fig. 6. Market Context & Price Forecasting Chart — boAt Airdopes 141 Gen 2 (Apr 2025 – Apr 2027)

D. System Stability and Scraping Reliability

The back-end did not crash once during all these test runs. Instead, failed connections were simply logged and ignored. Crawling went smoothly most of the time, particularly using ScraperAPI and random User-Agent headers, although we occasionally faced problems with bot detection.

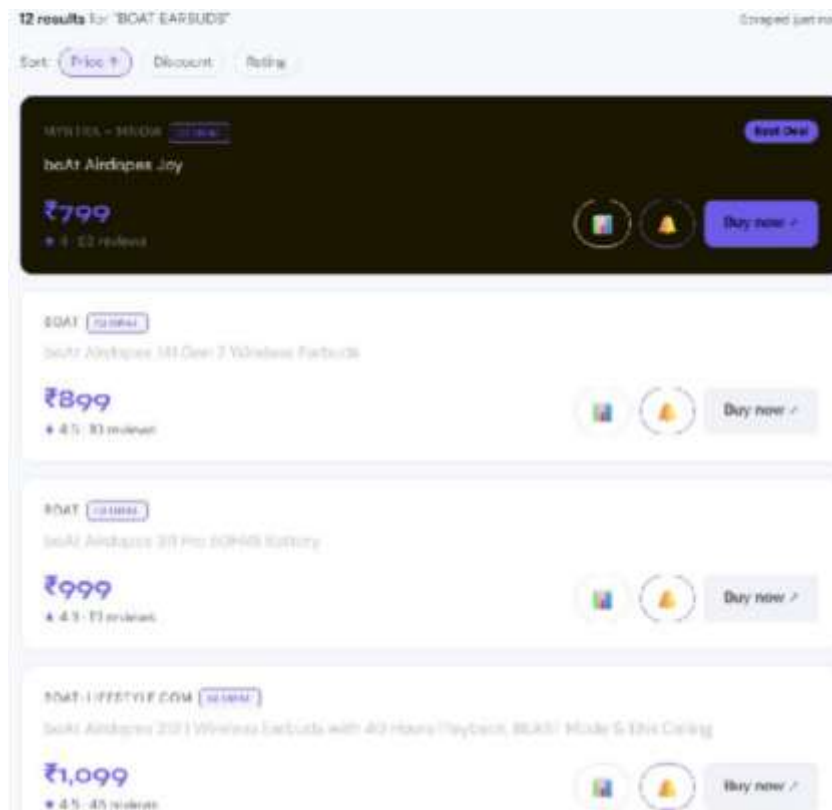


Fig. 7. Cross-Platform Price Comparison Results

E. Email Notification Delivery

The trigger-based emails reached inboxes quickly, taking only about 10 to 30 seconds. Trigger-based emails were effective in Gmail and Outlook. They included information about the product, price, and link to an offer..

VI. CONCLUSION AND FUTURE WORK

With SmartShopper, one can easily develop an efficient price prediction mechanism without being weighed down by elaborate frameworks or the use of headless browsers. With Node.js for backend development, Cheerio for web scraping, and user-agent switching, prices and alerts are in real-time. Google Gemini's artificial intelligence technology not only alerts the shoppers, but also offers them insights into price predictions for several months to come. In addition, through hourly checks and real-time notifications, the users will know when the ideal buying time comes.

Next is the development of a mobile application that will incorporate push notifications, advanced review analysis (both pricing and quality rating), shareable deal vaults for group monitoring, and ultimately, automation, whereby SmartShopper will automatically make purchases at the ideal prices.

REFERENCES

- [1] N. Kaur, "Price Prediction in E-Commerce Using Machine Learning Models: A Comparative Study " ShodhKosh: Journal of Visual and Performing Arts, vol. 4 No. 1 January to June 2023.
- [2] A. Abudurehman, Y. Zhao and A. Nilupaer "Design of E-Commerce Product Price Prediction Model Based on Generative Adversarial Network with Adaptive Weight Adjustment " Scientific Reports, vol. 15 Article 25126 July 2025.
- [3] M. Gupta, "Dynamic Pricing Optimization in E-Commerce Using AI and Machine Learning: A Comprehensive Framework for Demand Prediction and Revenue Maximization " SSRN, November 2024. You can find it here: <https://ssrn.com/abstract=5075196>
- [4] M. Nowak, "Dynamic Pricing Method in the E-Commerce Industry " Applied Sciences, vol. 14 No. 24 2024.

- [5] G.. Y. Horbunova, "Modeling and Forecasting Dynamic Factors of Pricing in E-Commerce " in Proc. CEUR Workshop. I-2020, Kyiv, Ukraine December 2020.
- [6] K. J. Ferreira, B. H. A. Lee and D. Simchi-Levi "Analytics for an Online Retailer: Demand Forecasting and Price Optimization " Manufacturing & Service Operations Management vol. 18 No. 1 2016. This study focuses on E-Commerce.
- [7] Y. J. Zhang and W. Wang, "Dynamic Pricing in E-Commerce Using Reinforcement Learning " IEEE Access, 2020. The authors explore E-Commerce.
- [8] A. A. Bukhari and S. Ahmad "Time Series Forecasting for Product Prices Using LSTM Networks " Journal of AI Research, 2021. They look at E-Commerce product prices.
- [9] T. Dhulipalla, N. Karumanchi, M. Munji, K. Harshitha and S. Kunchala "Price Comparison Application for E-Commerce Using Web Scraping " International Journal of Engineering Research 2025. The application is for E-Commerce.
- [10] H. Chen and K. Liu "Competitor-Aware Price Forecasting for E-Commerce Using Gradient Boosting Models " Expert Systems with Applications, 2022. This study is about E-Commerce.
- [11] S. Singh, G. Srivastava, V. Dubey and G. R. Mishra "Triggering an Email Alert Based on Price Comparison by Web Scraping Using Python " in Flexible Electronics for Electric Vehicles, Springer, 2024.
- [12] I. E. Okonkwo and O. Anya "Developing Products Update-Alert System for E-Commerce Websites Users Using HTML Data and Web Scraping Technique " arXiv preprint arXiv:2109.00656, 2021.
- [13] N. Patil et al. "Web Scraping for E-Commerce Websites " International Journal for Research in Applied Science and Engineering Technology (IJRASET) 2022.
- [14] P. Kyriakakis, T. Kalamboukis and S. Kotsiantis "A Novel Price Prediction Service for E-Commerce Categorical Data " Mathematics, vol. 11 No. 8 Article 1938 April 2023.
- [15] F. Chen, "Research on Real-Time E-Commerce Price Comparison System Using Python Web Scraping Technology " International Journal of Computer Science and Information Technology, vol. 4, No. 2 Pp. 127–136, 2024. The system is, for E-Commerce.