Int. j. adv. multidisc. res. stud. 2026; 6(6):1005-1010

Received: 04-10-2026
Accepted: 14-11-2026

International Journal of Advanced Multidisciplinary Research and Studies

ISSN: 2583-049X

Predictive Inventory Management System for Small Retail Shops Using Machine Learning

[1] Nkandu Femmy, [2] Jeyakumar V
[1] DMI-St. Eugene University, Chibombo Campus, Zambia
[2] Guide, DMI-St. Eugene University, Chibombo Campus, Zambia

Corresponding Author: **Nkandu Femmy**

Abstract
Inventory management is important for small retail shops because poor stock monitoring and inaccurate demand estimates can lead to product shortages, excess stock and losses caused by expired goods. This study presents a web-based predictive inventory management system developed to support stock control and sales demand prediction in small retail shops. The system integrates product management, stock batch monitoring, expiry tracking, sales processing, reporting and online customer ordering. Python, Pandas and Scikit-learn were used to implement and evaluate Linear Regression and Random Forest Regression. Their performance was compared using Mean Absolute Error and Root Mean Squared Error. Random Forest Regression achieved a mean absolute error of 1.033 and a root mean squared error of 1.198, while Linear Regression recorded 2.690 and 3.123, respectively. Random Forest Regression therefore produced lower prediction errors on the available evaluation dataset and was selected for the system's demand prediction component. The system also estimates the number of days before usable stock may run out and provides reorder recommendations based on predicted demand and available inventory. The study demonstrates how machine learning can be integrated with inventory management functions to support stock monitoring and replenishment decisions in small retail shops.

Keywords: Inventory Management, Machine Learning, Sales Demand Prediction, Random Forest Regression, Linear Regression, Small Retail Shops

1. Introduction

Inventory management is an important part of retail business operations. It involves monitoring and controlling stock to ensure that products are available when customers need them while reducing shortages, overstocking and losses. For small retail shops, effective inventory management is particularly important because products may have different sales rates, prices, suppliers and expiry periods.

Traditional inventory management methods mainly focus on recording products, monitoring stock quantities and managing sales transactions. Although these activities help businesses maintain stock records, they may not provide sufficient information about future product demand. When shop owners rely mainly on current stock levels or personal experience, they may find it difficult to identify products that are likely to run out or determine when replenishment is necessary. This can result in lost sales, excess stock and expiry-related losses.

Machine learning offers opportunities to improve inventory decision-making by analysing historical sales information and identifying patterns that can support demand prediction. Previous research has investigated the application of machine learning in inventory control and demand forecasting. Bergsma *et al*. (2025) [1] reviewed machine learning approaches in inventory control, while Douaioui *et al*. (2024) [2] examined machine learning and deep learning models for demand forecasting. Chen (2024) [3] investigated the use of Random Forest for retail sales prediction. Other studies have explored stockout prediction, retail demand forecasting and inventory replenishment using machine learning techniques (Liu *et al*., 2025; Walter *et al*., 2025; Singhal *et al*., 2024) [4, 5, 6].

These studies demonstrate the potential of data-driven methods to support demand forecasting and inventory decisions. However, small retail businesses may need practical systems that integrate predictive analysis with everyday inventory activities, including product management, sales recording, stock monitoring and expiry tracking.

This study addresses this need by developing a web-based Predictive Inventory Management System for Small Retail Shops Using Machine Learning. The system combines conventional inventory management functions with machine-learning-based demand prediction. It also includes stock batch management, expiry monitoring, sales reporting, online customer ordering and stock discrepancy detection.

The machine learning component compares Linear Regression and Random Forest Regression using Mean Absolute Error and Root Mean Squared Error. The selected model is used to predict product demand. The predicted demand is then combined with current usable stock to estimate potential stockout periods and generate reorder recommendations.

The main aim of this study is to design and develop an integrated predictive inventory management system that supports small retail shops in managing stock, predicting product demand and making informed replenishment decisions.

2. Materials and Methods

2.1 Research and Development Approach

This study adopted a system development approach to design and implement a predictive inventory management system for small retail shops. The development process involved identifying inventory management requirements, designing the system, implementing its main functions and testing the developed application. Machine learning techniques were incorporated to analyse historical sales data, predict product demand and support stock replenishment decisions.

2.2 System Development Technologies

The system was developed using PHP for server-side processing and MySQL for database management. HTML, CSS, Bootstrap and JavaScript were used to develop the user interface. Python was used for machine learning implementation, while Pandas supported sales data preparation and Scikit-learn provided the regression algorithms. The application was developed and tested using the XAMPP local server environment. Visual Studio Code was used as the development editor.

2.3 System Architecture and Functional Modules

The system connects the web application, MySQL database and Python-based machine learning component. PHP manages communication between the application and the database, while Python processes historical sales information and generates demand predictions. The prediction results are saved in a JSON file, which the PHP application reads and displays on the machine learning predictions page. The system includes product management, product variant management, stock batch monitoring, expiry tracking, sales processing, sales reporting, online customer ordering and stock discrepancy detection. Role-based access is provided for administrators, cashiers and customers.

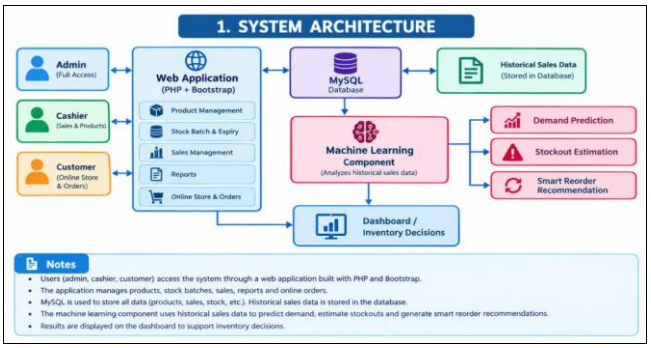


Fig 1: System architecture of the predictive inventory management application

2.4 Data Collection and Preparation

Historical sales information was retrieved from the system's MySQL database. The model evaluation process used records from the sales, saleitems and products tables. These records contained sales dates, product names and quantities sold. The sales dates were converted into date values, and sales quantities were grouped by product and date to calculate daily sales. Four input features were then extracted: product code, day of the month, month and day of the week. Daily sales quantity was used as the target variable for demand prediction. For the final prediction component, the data retrieval process was extended to include product variants. Standard products and individual variants were identified separately using product and variant identifiers. Sales quantities were grouped by item and date so that demand could be predicted for individual products and variants. The preparation process used the recorded sales dates and did not insert additional records for dates without recorded sales. Therefore, the dataset represents the available recorded sales information.

2.5 Machine Learning Models and Evaluation

Two regression algorithms, Linear Regression and Random Forest Regression, were implemented and compared during the model evaluation stage. The prepared daily sales records were divided into training and testing datasets using an 80:20 proportion. The first 80% of the grouped records were used to train the models, while the remaining 20% were used for testing. This evaluation was performed when at least five grouped daily sales records were available. Both models used product code, day, month and day of the week as input features, with daily sales quantity as the prediction target. Model performance was assessed using Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). These metrics measured the differences between actual sales quantities and predicted quantities. Lower values indicate smaller prediction errors. Random Forest Regression was configured with 100 estimators and a random state of 42. Based on the evaluation results recorded in the study, Random Forest Regression achieved lower MAE and RMSE values than Linear Regression and was selected for the final demand prediction component.

2.6 Demand Prediction and Inventory Analysis

The final prediction component was implemented in Python using the Random Forest Regression algorithm. The model retrieves available historical sales data, prepares daily sales records and trains the model using the available dataset. It then predicts demand for the day following the latest recorded sales date. The final prediction script uses product and variant identifiers to distinguish standard products from individual product variants. Its input features are item code, day, month and day of the week, while daily sales quantity is the prediction target. The system also retrieves current stock quantities from stock batch records. Expired stock is excluded by considering only batches whose expiry dates are on or after the current date. This ensures that the stock quantity used for stockout estimation represents usable inventory. The estimated number of days until stockout is calculated as: Estimated days until stockout = Current usable stock / Predicted daily demand. When predicted demand is greater than zero, the system calculates the estimated stockout period. When there is no positive demand prediction, a numerical stockout estimate is not generated.

2.7 Reorder Recommendation Method

The system generates reorder recommendations by considering current usable stock, the product's reorder level and the estimated stockout period. 'Reorder Now' is generated when current usable stock is equal to or below the defined reorder level. 'Reorder Soon' is generated when stock is above the reorder level but the estimated stockout period is seven days or less. 'Stock Level OK' is generated when stock is above the reorder level and the estimated stockout period is greater than seven days. 'No Sales History' is displayed when no matching historical sales records are available and current stock is above the reorder level. These recommendations are intended to assist shop owners in identifying products that may require replenishment.

2.8 Inventory Control and System Testing

The system incorporates stock batch monitoring and First-Expired, First-Out (FEFO) inventory handling to support the management of products with different expiry dates. Expired stock is excluded from normal usable inventory calculations, while expiry monitoring helps administrators identify products approaching or exceeding their expiry dates. Stock discrepancy detection compares recorded inventory quantities with physical stock counts. The system displays identified differences to support further investigation by the administrator. Functional testing was conducted on the main system modules, including user authentication, product management, stock batch management, sales processing, online customer ordering, machine learning predictions, reporting and stock discrepancy detection. The tests assessed whether the implemented functions operated as expected under the tested conditions.

3. Results and Discussion

3.1 System Implementation Results

The study resulted in the development of a web-based predictive inventory management system for small retail shops. The system integrates inventory management functions with machine-learning-based demand prediction. The implemented application provides separate access for administrators, cashiers and customers. The administrator can manage products, product variants, stock batches, sales reports, machine learning predictions and customer orders. Cashiers can process sales and perform permitted order-related activities, while customers can register, log in, browse products and place online orders. The system also provides stock batch monitoring, expiry tracking, stock discrepancy detection and sales reporting. These functions allow shop owners to monitor inventory activities and access information that supports stock management decisions.

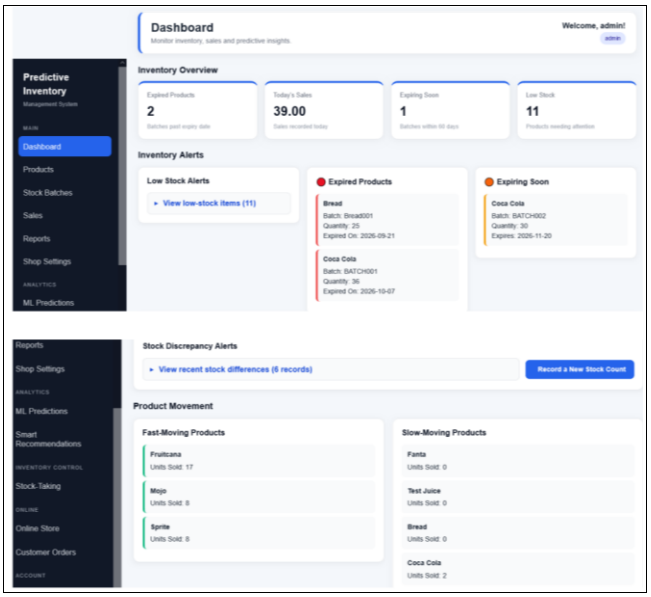


Fig 2: Administrator dashboard showing inventory overview and shop activity

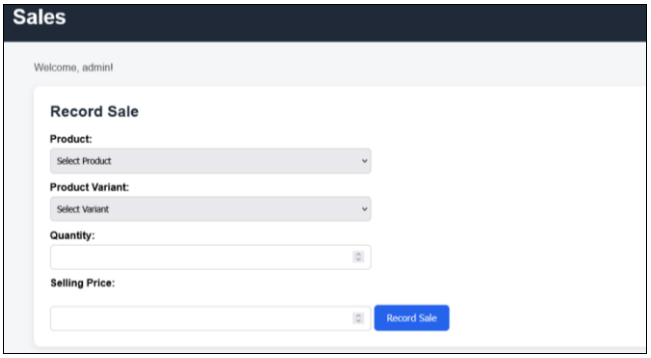


Fig 3: Sales processing interface for recording product and variant transactions

3.2 Inventory Management and Expiry Monitoring

The inventory management module supports product and product variant management, stock batch recording and stock quantity monitoring. The system records expiry dates for stock batches and identifies expired and approaching-expiry products. The First-Expired, First-Out (FEFO) approach is used when selecting stock for sales processing. This helps ensure that products with earlier expiry dates are prioritised over later-expiring stock. Expired stock is excluded from usable inventory calculations and normal stock allocation. The stock discrepancy detection function compares recorded inventory quantities with physical stock counts. Differences are displayed to the administrator for further investigation. This provides an additional means of

monitoring inventory accuracy without automatically treating a discrepancy as evidence of theft or misconduct.

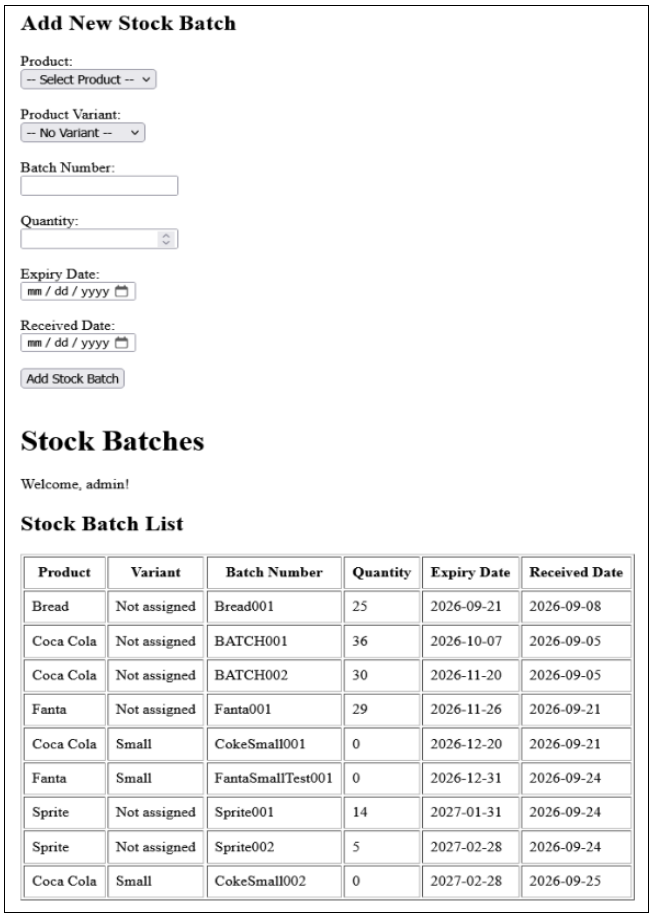


Fig 4: Stock batch management interface showing batch quantities and expiry dates

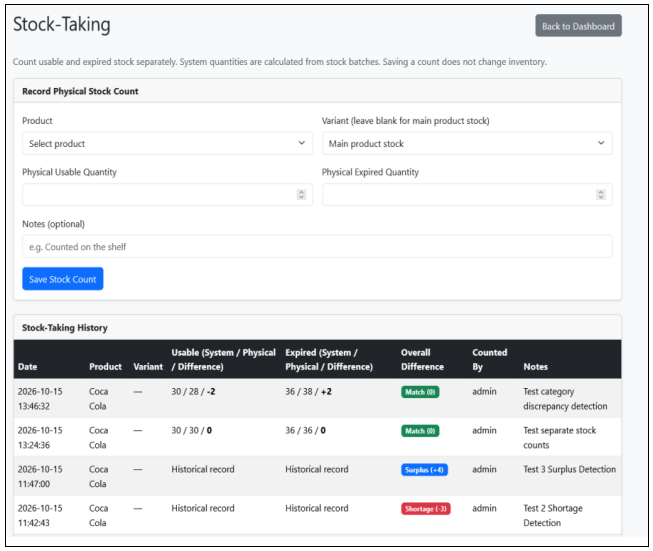


Fig 5: Stock-taking interface showing physical counts and recorded stock discrepancies

3.3 Sales Reporting Results

The sales reporting module provides information about sales performance over selected periods. It displays total sales value, quantities sold, number of transactions and average sale value. The figures below were recorded during system testing.

Table 1

Report period	Total sales	Units sold	Transactions	Average sale
All Time	K1,023.00	80	62	K16.50
This Month	K243.00	23	21	K11.57
Custom range (24–29 September 2026)	K420.00	34	24	—

The results demonstrate that the reporting module can summarise recorded sales information and support the monitoring of shop transactions. Users can also apply date filters to examine sales over selected periods.

3.4 Comparison of Machine Learning Models

Linear Regression and Random Forest Regression were evaluated using Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). The recorded evaluation results are presented in Table 2.

Table 2

Machine learning model	MAE	RMSE
Linear Regression	2.690	3.123
Random Forest Regression	1.033	1.198

The results show that Random Forest Regression achieved lower error values than Linear Regression. Its MAE of 1.033 and RMSE of 1.198 were lower than the corresponding Linear Regression values of 2.690 and 3.123. This indicates that Random Forest Regression produced smaller prediction errors on the evaluation dataset used in this study. Therefore, it was selected for the final demand prediction component. However, these results should be interpreted in relation to the available sales dataset and the evaluation procedure used. They do not establish that the model will achieve the same performance across all retail businesses or larger datasets.

3.5 Product Demand Prediction and Stockout Estimation

The implemented Random Forest model predicts daily product demand using historical sales information. The predicted demand is combined with current usable stock to estimate how many days the available inventory may last. Table 3 presents selected prediction results recorded during system testing.

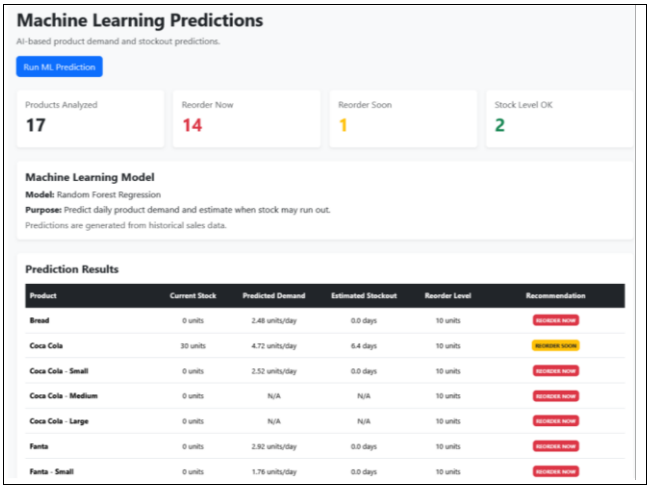


Fig 6: Machine learning predictions interface showing demand and reorder recommendations

Table 3

Product	Current stock (units)	Predicted demand (units/day)	Estimated days until stockout	Recommendation
Coca Cola	66	2.30	28.7	Stock Level OK
Bread	22	4.33	5.1	Reorder Soon
Fanta	29	6.39	4.5	Reorder Soon

The results show that the system generates different recommendations based on predicted demand and available stock. Coca Cola had sufficient stock for the estimated demand period and was classified as Stock Level OK. Bread and Fanta had shorter estimated stockout periods and were classified as Reorder Soon. An additional test was conducted by reducing Coca Cola's available stock to eight units. The system estimated that the remaining stock would last approximately 3.0 days and generated a Reorder Now recommendation. This demonstrates how changes in stock quantities can affect the recommendations produced by the system. Products and variants without matching sales history are handled separately. In such cases, the system does not generate a demand estimate, but it can still recommend replenishment when current stock is at or below the defined reorder level.

3.6 Online Ordering and Stock Updates

The online store allows customers to browse available products and submit orders through the web application. The system processes customer orders and updates inventory records when orders are successfully completed. During testing, online order number 29 was processed successfully and generated sales record number 64. This demonstrated the integration between customer ordering, sales recording and inventory updates. The integration helps maintain consistency between online transactions and the shop's sales and stock records.

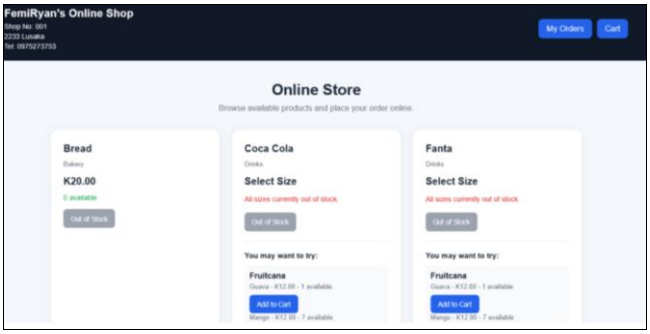


Fig 7: Online store interface used by customers to browse products and select variants

3.7 Functional Testing Results

Functional testing was conducted to assess the operation of the main system modules. The tested areas included user authentication, product and stock management, sales processing, customer orders, machine learning predictions, reporting and stock discrepancy detection. The recorded tests indicated that the major functions operated as expected under the test conditions. The machine learning module generated prediction results, while the reporting module displayed sales summaries and supported date filtering. Stock discrepancy detection also identified differences between recorded and physically counted stock without automatically changing inventory quantities. These results demonstrate that the developed application integrates its main inventory management and predictive functions. However, the testing was conducted within the developed system environment and does not establish performance under all possible retail operating conditions.

3.8 Discussion

The findings demonstrate that machine learning can be integrated into a web-based inventory management application to support demand prediction and replenishment decisions. The comparison between Linear Regression and Random Forest Regression showed that Random Forest achieved lower prediction errors on the available evaluation dataset. The system extends basic inventory recording by combining historical sales analysis with current usable stock quantities. Its stockout estimates and reorder recommendations provide shop owners with additional information when deciding which products may require replenishment. The integration of expiry monitoring, FEFO stock handling, sales reporting, online ordering and discrepancy detection also provides inventory management functions within a single application. Nevertheless, the study has limitations. The machine learning models were evaluated using the available sales records, which may not represent all seasonal or changing retail demand patterns. The evaluation script grouped sales by product name, while the final prediction script separately handled product variants. In addition, the prediction features were limited to item identification and calendar information. Factors such as promotions, weather, supplier delays and seasonal events were not included. Further evaluation using larger datasets, additional retail shops and more comprehensive validation procedures would help establish the reliability and wider applicability of the proposed approach.

Note: The figures illustrate the system interfaces captured during project testing. Values displayed in the screenshots represent test snapshots and may differ from the summarized evaluation results.

4. Conclusion

This study developed a web-based Predictive Inventory Management System for Small Retail Shops Using Machine Learning. The system integrates product management, stock batch monitoring, expiry tracking, sales processing, reporting, online customer ordering and stock discrepancy detection. The study implemented and compared Linear Regression and Random Forest Regression using Mean Absolute Error and Root Mean Squared Error. The evaluation results showed that Random Forest Regression achieved lower prediction errors, with an MAE of 1.033 and an RMSE of 1.198, compared with Linear Regression, which recorded an MAE of 2.690 and an RMSE of 3.123. Based on these results, Random Forest Regression was selected for the final demand prediction component. The developed system uses historical sales information to predict daily product demand and combines the predictions with available usable stock to estimate potential stockout periods. It also generates reorder recommendations to assist shop owners in making replenishment decisions. Functional testing demonstrated that the major system modules operated as expected under the tested conditions. The study therefore demonstrates the practical integration of machine learning with inventory management functions to support stock monitoring and decision-making in small retail shops.

Although the results are promising, the study was limited by the available sales dataset and the features used for prediction. Further development and evaluation using larger datasets and additional retail environments would help improve the system and assess its wider applicability.

5. Recommendations

1. Use larger sales datasets: Small retail shops should maintain accurate historical sales records to provide more information for training and evaluating demand prediction models.
2. Include additional prediction factors: Future versions should consider seasonal changes, promotions, weather conditions and other factors that may influence customer demand.
3. Evaluate additional machine learning models: Other regression and time-series forecasting techniques could be investigated to determine whether they improve prediction performance.
4. Improve system accessibility: Future development could include secure online hosting and improved mobile accessibility so that shop owners can monitor inventory from different locations.
5. Conduct wider system testing: Further testing should involve additional retail shops and larger datasets to assess the system's reliability and suitability in different business environments.

6. Acknowledgement

The authors would like to express their appreciation to DMI-St. Eugene University for providing the academic environment and support required for this study. Special appreciation is extended to the project guide, Jeyakumar V., for his guidance and support throughout the development of the project.

7. References

1. Bergsma R, De Ruijt C, Bhulai S. A systematic review of machine learning approaches in inventory control optimization. Operations Research Perspectives. 2025; 15:100367. Doi: 10.1016/j.orp.2025.100367
2. Douaioui K, Oucheikh R, Benmoussa O, Mabrouki C. Machine learning and deep learning models for demand forecasting in supply chain management: A critical review. Applied System Innovation. 2024; 7(5):93. Doi: 10.3390/asi7050093
3. Chen D. Walmart sales prediction based on random forest model and application of feature importance. Applied and Computational Engineering. 2024; 53:264-273. Doi: 10.54254/2755-2721/53/20241461
4. Liu Y, Kalaitzi D, Wang M, Papanagnou C. A machine learning approach to inventory stockout prediction. Journal of Digital Economy. 2025; 4:144-155. Doi: 10.1016/j.jdec.2025.06.002
5. Walter A, Ahsan K, Rahman S. Application of artificial intelligence in demand planning for supply chains: A systematic literature review. The International Journal of Logistics Management. 2025; 36(3):672-719. Doi: 10.1108/IJLM-02-2024-0120
6. Singhal K, Singh V, Kaul A. Smart retail: Utilizing machine learning for demand prediction, price strategy, and inventory management. In: 2024 IEEE 16th International Conference on Computational Intelligence and Communication Networks (CICN), 2024, 485-489. Doi: 10.1109/CICN63059.2024.10847534