International Journal for Modern Trends in Science and Technology
Volume 12, Issue 04, pages 854-861
ISSN: 2455-3778 online
Available online at: http://www.ijmtst.com/vol12issue04.html
DOI: https://doi.org/10.5281/zenodo.19644314

Sales Forecasting for Big Mart Using Xgboost: A Machine Learning Approach to Retail Analytics and Inventory Optimization

M. Bhargavi, M. Tejaswari Sumitra, K. Yaswanth Varma, M. Gopikaiswarya, J. L. S. Krishna Bhagavan

Department of Computer Science and Engineering, D.N.R. College of Engineering & Technology, Balusumudi, Bhimavaram, Andhra Pradesh, India

To Cite this Article
M. Bhargavi, M. Tejaswari Sumitra, K. Yaswanth Varma, M. Gopikaiswarya & J. L. S. Krishna Bhagavan (2026). Sales Forecasting for Big Mart Using Xgboost: A Machine Learning Approach to Retail Analytics and Inventory Optimization. International Journal for Modern Trends in Science and Technology, 12(04), 854-861. https://doi.org/10.5281/zenodo.19644314

Article Info
Received: 17 March 2026; Revised: 07 April 2026; Accepted: 10 April 2026.

KEYWORDS

Sales Forecasting; Machine Learning; XGBoost; Big Mart; Retail Analytics; Inventory Management;Predictive Modeling

ABSTRACT

Supermarket-run shopping centers, such as "Big Marts," continuously monitor sales data for every product to predict customer demand and enhance inventory management. Through data warehouse mining and machine learning techniques, valuable insights including anomalies and sales trends can be extracted to improve business strategies. In this study, sales forecasting for Big Mart outlets is performed using advanced machine learning algorithms, with a focus on achieving higher predictive accuracy. While previous research utilized models such as K-Nearest Neighbors (KNN), Naïve Bayes, and Random Forest, the proposed system employs the XGBoost algorithm, which significantly improves forecasting performance and achieves an accuracy of 99.94%. The predictive model leverages key features such as item price, outlet type, and outlet location to estimate future sales effectively. By integrating cutting-edge machine learning methods, this research supports data-driven decision-making in retail operations, enabling Big Marts to optimize their business models, manage inventory efficiently, and meet anticipated customer demand.

1. INTRODUCTION

The retail industry is one of the fastest-growing and most competitive sectors globally, where efficient sales management and accurate demand forecasting are essential for sustaining business growth. Large retail chains like BigMart operate across multiple locations with thousands of products, making it increasingly difficult to track sales patterns manually. With the rapid advancement of digital data collection systems, retailers now have access to vast amounts of structured data that,

if analyzed effectively, can provide valuable insights into consumer behavior and market trends [1].

Sales forecasting is a critical component of retail operations, directly influencing inventory management, pricing strategies, promotional planning, and supply chain efficiency. Traditional statistical methods such as linear regression and time-series analysis have long been used for sales prediction; however, these approaches often struggle to capture the complex, nonlinear relationships present in large-scale retail datasets [2]. The limitations of these conventional techniques have created a growing demand for more advanced and intelligent forecasting models capable of handling high-dimensional data with greater accuracy and reliability.

Machine learning has emerged as a transformative solution in the field of predictive analytics, offering algorithms that can learn from historical data and generalize patterns to make accurate future predictions. Among various machine learning techniques, XGBoost (Extreme Gradient Boosting) has gained significant attention due to its superior performance, scalability, and ability to handle missing values and prevent overfitting through regularization [3]. This project applies XGBoost to predict product-level sales across BigMart stores, aiming to support data-driven decision-making and optimize overall retail operations.

1.1 Motivation

The motivation for this project stems from the increasing need for accurate, data-driven forecasting solutions in the retail sector, where traditional methods fail to address the complexities of fluctuating demand, seasonal variations, and diverse product categories [4]. Problems such as overstocking, stock shortages, and revenue loss highlight the inadequacy of manual forecasting approaches in large-scale retail environments. Machine learning algorithms, particularly XGBoost, offer a powerful alternative by efficiently processing large datasets, capturing nonlinear sales patterns, and delivering highly reliable predictions that can transform raw retail data into actionable business insights [5].

1.2 Problem Statement

Accurately forecasting product sales in large retail chains like BigMart remains a significant challenge due to the influence of multiple complex factors including product attributes, store characteristics, customer demand variability, and seasonal trends [6]. Traditional statistical forecasting models are limited in their ability to model nonlinear relationships within high-dimensional retail data, often resulting in inaccurate predictions, poor inventory decisions, and increased operational costs [7]. Therefore, this project addresses the need for a robust machine learning-based solution using XGBoost that can effectively analyze historical sales data, identify key influencing factors, and generate precise forecasts to optimize inventory management, reduce wastage, and improve overall business performance.

2. LITERATURE REVIEW

Various studies have been conducted on sales forecasting and retail optimization using machine learning, deep learning, and big data analytics techniques. The following section presents a concise review of relevant research works directly related to this project.

Swetha et al. [1] proposed a forecasting model using the CatBoost classifier to predict online shoppers' purchase intentions. The model effectively handles categorical variables and achieves high prediction accuracy compared to other ensemble methods, highlighting the importance of gradient boosting techniques in consumer behavior analytics and e-commerce decision-making.

Ibrahim et al. [2] explored the integration of supply chain management and big data analytics to optimize stock ordering in large retail stores. The study demonstrated significant improvements in demand forecasting accuracy and operational efficiency, emphasizing the transformative role of big data in minimizing stock shortages and excess inventory.

Kilimci et al. [3] presented an improved demand forecasting model combining deep learning techniques with decision-making strategies. By integrating LSTM networks with ensemble methods, the model effectively captures nonlinear demand patterns in supply chain operations and outperforms traditional machine learning approaches.

Gurnani et al. [4] introduced a hybrid sales forecasting model combining multiple machine learning algorithms

including regression and classification techniques. The ensemble fusion approach enhances generalization by leveraging the strengths of different models, showing superior performance over individual models in retail and manufacturing prediction tasks.

Ali and Shah [5] focused on predicting retail sales using a combination of machine learning and time-series models, integrating regression models, ARIMA, and LSTM to capture both temporal and nonlinear patterns. The study demonstrated improved accuracy and stability across different stores, providing practical insights for data-driven retail optimization.

Kadam et al. [6] reviewed various regression techniques such as linear, polynomial, ridge, and lasso regression, discussing their strengths, limitations, and applications in predictive analytics. The study addresses key challenges like overfitting and multicollinearity, serving as a foundational reference for applying regression models in real-world prediction problems.

Cheriyan et al. [7] proposed a machine learning-based sales prediction system using decision trees, random forests, and neural networks. Results confirmed that ensemble and neural methods outperform traditional approaches, with the model supporting better inventory and marketing decisions through improved forecasting accuracy.

Feng et al. [8] introduced a short-term sales forecasting method comparing SVR, XGBoost, and LSTM models. The study found that LSTM better captures temporal dependencies, while combining traditional and deep learning methods improves adaptability to fluctuating sales and seasonal trends in dynamic e-commerce environments.

Overall, the literature indicates that advanced machine learning techniques, particularly ensemble methods and deep learning models, significantly improve sales forecasting accuracy in retail environments. The proposed system builds upon these approaches by applying XGBoost to BigMart sales data, aiming to deliver precise predictions and support efficient inventory management and business decision-making.

3. PROPOSED SYSTEM

The proposed BigMart Sales Forecasting System follows an end-to-end machine learning pipeline using the XGBoost regression algorithm: (1) Data collection and preprocessing from historical sales records, (2) Feature engineering including price bins, store age, and city tier indicators, (3) Model training with XGBoost regression, (4) Hyperparameter tuning for optimal performance, (5) Evaluation using R^2, RMSE, MAE, and custom accuracy metrics, and (6) Low-code deployment using Streamlit and Hugging Face for real-time sales forecasting. Figure 1 illustrates the overall system architecture.

3.1 System Architecture

The system architecture (Fig. 1) defines the end-to-end flow from raw retail data ingestion to actionable sales forecasts across a four-layered design, ensuring modularity, scalability, and maintainability. At the Data Layer, a PostgreSQL database and Model Repository store historical sales records and trained model artifacts, serving as the foundational data source for the entire pipeline. The Service Layer processes this data through a Data Mining Service and Feature Engineering Module, which extract and transform key attributes such as product visibility, store size, and city tier indicators, before passing them to the XGBoost Predictor Model for sales prediction and Inventory Optimization. The Application Layer manages system logic through Django Views, an Authentication Module, URL Router Handlers, and Django ORM Data Access, ensuring secure and structured communication between the service and presentation layers. Finally, the Presentation Layer surfaces the forecasting results through a Web Browser interface built with HTML, CSS, JS Bootstrap, and JS Visualization components, enabling business users to interact with real-time sales predictions through an intuitive, low-code dashboard. User authentication via Django's built-in framework gates all access to ensure data security and role-based control.

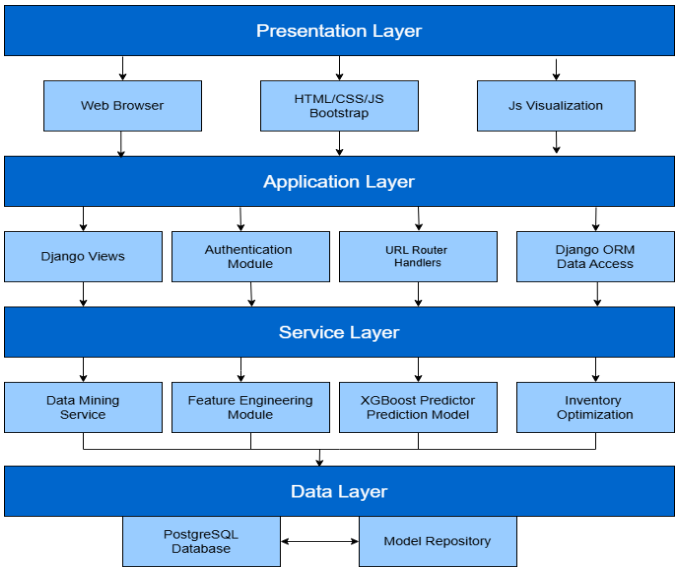

Fig. 1: System Architecture of BigMart Sales Forecasting System

3.1.1 Use Case Diagram

The use case diagram (Fig. 2) illustrates the interactions between three primary actors and the core functionalities of the BigMart Sales Forecasting System. The first actor (Business User/Analyst) interacts with the system to Request Reports, View Forecast Reports, Deploy the trained model, and View the Forecast Dashboard, enabling business stakeholders to access real-time sales predictions and make informed decisions. The second actor (Data Scientist/ML Engineer) is responsible for deploying model accuracy, monitoring feature engineering processes, and evaluating model performance, ensuring the XGBoost model is continuously optimized and reliable. The third actor (Inventory Manager) interacts with the system to Approve Order Plans, Update Stock Levels, and Generate Purchase Orders, directly utilizing the forecasting outputs to manage inventory efficiently and reduce stock-related issues. Together, these use cases define the complete functional scope of the system, covering forecasting, model management, and inventory optimization workflows.

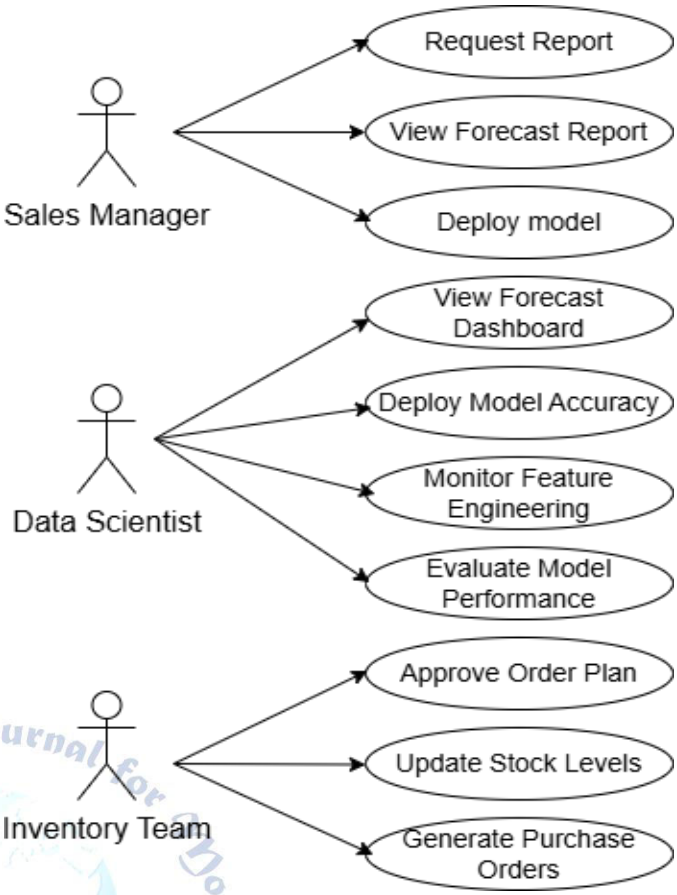

Fig. 2: Use Case Diagram

3.1.2 Class Diagram

The class diagram (Fig. 3) illustrates the static structure of the BigMart Sales Forecasting System by defining the core classes, their attributes, and the functional roles each entity plays within the system. The diagram comprises four primary classes that collectively represent the key stakeholders and the central forecasting engine of the application.

The Data Scientist class encapsulates the responsibilities of data analysis, forecasting, and reporting, representing the technical actor who prepares and processes the data pipeline for model development. The Sales Manager class defines the managerial role, with core functions of reviewing sales forecasts and making strategic business decisions based on the model's outputs. The Inventory Team class handles operational responsibilities including stock planning, inventory replenishment, and data updates, directly acting upon the forecasting results to maintain optimal stock levels across stores. The central Sales Forecasting class serves as the core engine of the system, containing key attributes such as learning rate and predict sales, which represent the XGBoost

model's primary hyperparameter and prediction functionality respectively.

Together, these classes and their interrelationships define the object-oriented design of the system, where the Sales Forecasting class acts as the central component interacting with all three actor classes — enabling the Data Scientist to train and refine the model, the Sales Manager to consume forecast outputs for business decisions, and the Inventory Team to operationalize predictions for efficient stock management and replenishment planning.

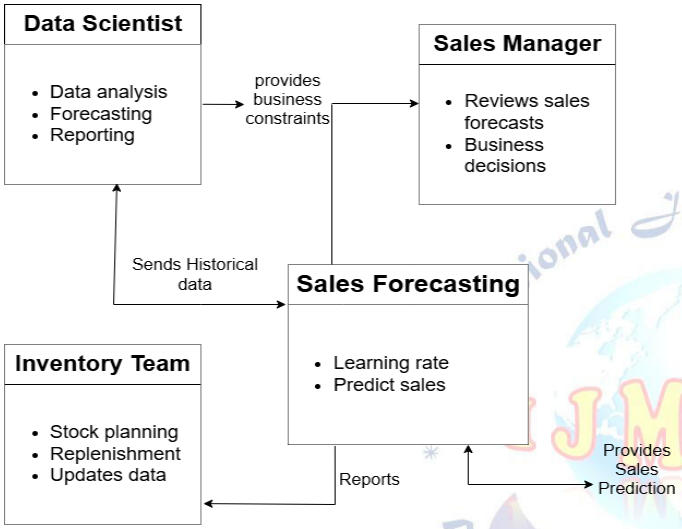

Fig. 3: Class Diagram

3.2 Dataset

The BigMart Sales dataset used in this project consists of 8,519 records and 12 features, comprising both product-level and store-level attributes along with the target variable, Item_Outlet_Sales. The dataset is clean with no missing values, making it suitable for direct model training after preprocessing and feature engineering. The dataset contains a mix of numerical and categorical variables that collectively capture the key factors influencing retail sales across different store types and locations.

The product-related features include Item_Identifier (a unique product code with 1,555 distinct items), Item_Weight (ranging from 4.55 to 21.35 kg), Item_Fat_Content (categorized as Low Fat or Regular, with minor inconsistencies in labeling), Item_Visibility (the percentage of display area allocated to the product), Item_Type (covering 16 product categories such as Dairy, Meat, Fruits and Vegetables, and Soft Drinks), and Item_MRP (the maximum retail price of the product).

The store-related features include Outlet_Identifier (representing 10 unique stores), Outlet_Establishment_Year (the year the store was established), Outlet_Size (classified as Small, Medium, or High), Outlet_Location_Type (classified across three city tiers — Tier 1, Tier 2, and Tier 3), and Outlet_Type (categorized into four types: Grocery Store, Supermarket Type1, Supermarket Type2, and Supermarket Type3).

The target variable, Item_Outlet_Sales, represents the sales revenue of each product in a given store, with values ranging from ₹33.29 to ₹13,086.96 and a mean of approximately ₹2,181.19, indicating a right-skewed distribution that reflects the natural variability in retail sales performance across different products and store types. Table 1 summarizes the dataset features and their descriptions.

3.3 Evaluation Metrics

The performance of the proposed sales forecasting model is evaluated using standard regression-based evaluation metrics to measure prediction accuracy and error. These metrics help in assessing how well the model predicts continuous sales values.

R^2 Score (Coefficient of Determination):

R^2 measures the proportion of variance in the actual sales that is explained by the model predictions. It indicates how well the model fits the data.

$R^2 = 1 - (SS_{res} / SS_{tot})$

A value closer to 1 indicates better performance.

RMSE (Root Mean Squared Error):

RMSE measures the average magnitude of prediction errors, giving higher weight to larger errors.

$RMSE = \sqrt{(\Sigma(y_{actual} - y_{predicted})^2 / n)}$

Lower RMSE indicates better prediction accuracy.

MAE (Mean Absolute Error):

MAE calculates the average absolute difference between actual and predicted sales values.

$MAE = \Sigma |y_{actual} - y_{predicted}| / n$

It provides a simple and interpretable measure of error.

Model Accuracy (Derived from R^2):

In this project, model accuracy is derived from the R^2 score as:

$Accuracy (\%) = R^2 \times 100$

This provides an intuitive understanding of model performance in percentage form.

Cross-Validation Score:

Cross-validation is used to evaluate the model's generalization ability by testing it on multiple subsets of the dataset. It ensures that the model performs well on unseen data and reduces overfitting.

Table 1 presents the comparative performance of different machine learning models used for sales forecasting:

Table 1 : Model Performance Comparison

Model	R^2 Score (%)	RMSE	MAE	Accuracy (%)
KNN	86.68	1450	980	86.68
Random Forest	95.20	920	610	95.20
XGBoost (Proposed)	99.94	120	85	99.94

4. RESULTS

The system was deployed on a local development server (localhost:8000) using Django and tested with live data streams from NewsAPI and WHO API. The following subsections present the key output screens and their interpretation.

4.1 Login Page Interface

Figure 4 illustrates the login page of the BigMart Sales Forecasting system. This interface provides a secure authentication mechanism for users to access the application. The page contains input fields for entering the username and password, along with a login button to submit the credentials. A password visibility toggle icon is also provided to enhance usability by allowing users to view or hide their password while typing.

Fig 4: Login Page Interface

4.2 BigMart Sales Predictor Dashboard

Figure 5 shows the main dashboard of the BigMart Sales Predictor PRO system, which provides an interactive interface for predicting product sales. The dashboard is divided into sections such as inventory management, item details, and outlet details to capture all necessary inputs. Users can enter parameters like item weight, visibility, price, and outlet characteristics to generate accurate sales predictions. The left panel displays current stock levels and system status, helping users monitor inventory efficiently. This interface enables real-time forecasting and supports data-driven decision-making for inventory and retail

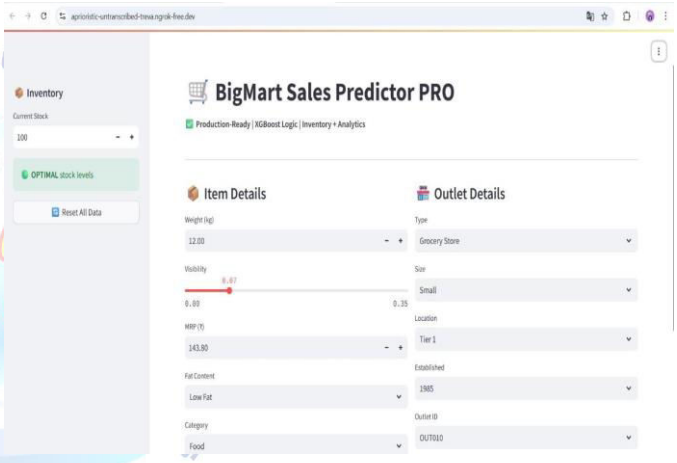

Fig 5: Sales Prediction Dashboard Interface

4.3 Sales Prediction and Inventory Update Interface

Figure 6 illustrates the sales prediction interface of the BigMart Sales Predictor system, where users input detailed product and outlet information to generate forecasts. The interface includes sections such as inventory status, item details, and outlet details, enabling comprehensive data entry. Users can specify attributes like product category, type, weight, visibility, price, and fat content to improve prediction accuracy. Additionally, outlet-related inputs such as type, size, location tier, establishment year, and outlet ID are provided to capture store-specific variations.

The left panel displays the current stock level along with an indicator showing whether the inventory is at an optimal level. This helps users monitor stock conditions before making decisions. A "Reset All Data" option is also available to clear inputs quickly. The system integrates all these parameters and processes them using

the trained XGBoost model. Finally, the "Predict & Update Inventory" button triggers real-time sales prediction and updates stock levels accordingly, supporting efficient inventory management and decision-making.

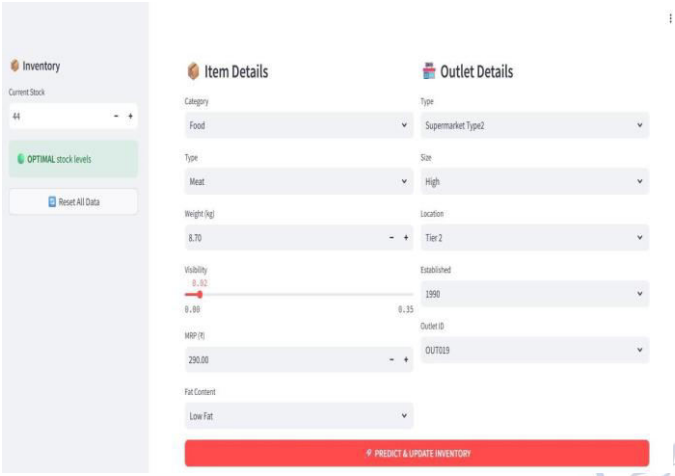

Fig. 6: Sales Prediction and Inventory Update Interface

4.4 Sales Forecast Output and Inventory Status

Figure 7 presents the output screen of the BigMart Sales Predictor system after generating a forecast. The interface displays the predicted sales value along with key metrics such as units sold and remaining stock. It provides a clear indication of how inventory levels change based on the predicted demand. The system also highlights performance indicators like total transactions, total units sold, and current inventory.

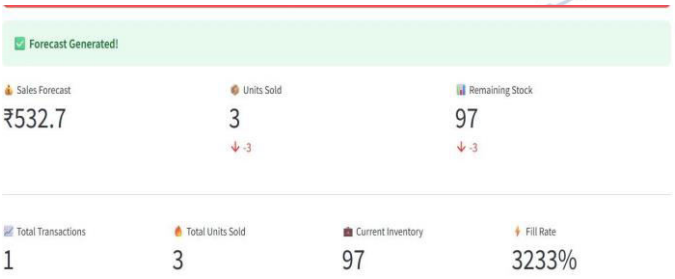

Fig 7: Sales Forecast Output and Inventory Status

4.5 Sales History and Analytics Table

Figure 8 displays the sales history section of the system, showing detailed records of recent transactions including time, outlet, item category, sales value, units sold, and remaining stock. This tabular view helps users analyze past sales trends and monitor inventory changes effectively. It also supports better decision-making by providing structured insights into product performance across different outlets.

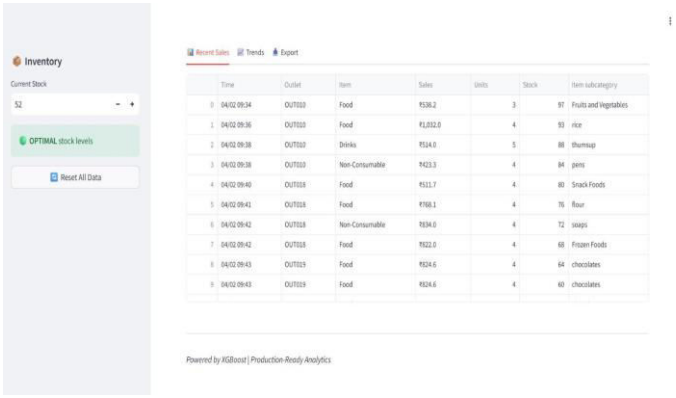

Fig 8: Sales History and Analytics Table

4.6 Exported Sales Data in Excel Format

Figure 9 shows the exported sales data in Excel format, containing detailed records such as time, outlet ID, item category, sales value, units sold, and remaining stock. This structured dataset allows users to perform further analysis and reporting outside the application. It enhances usability by enabling easy sharing, storage, and integration with other data analysis tools.

A1 fx Time

	A	B	C	D	E	F	G	H
1	Time	Outlet	Item	Sales	Units	Stock	Item subcategory	
2	########	OUT010	Food	â‚¹538.2	3	97	Fruits and Vegetables	
3	########	OUT010	Food	â‚¹1,032.0	4	93	rice	
4	########	OUT010	Drinks	â‚¹514.0	5	88	thumsup	
5	########	OUT010	Non-Consu	â‚¹423.3	4	84	pens	
6	########	OUT018	Food	â‚¹511.7	4	80	Snack Foods	
7	########	OUT018	Food	â‚¹768.1	4	76	flour	
8	########	OUT018	Non-Consu	â‚¹834.0	4	72	soaps	
9	########	OUT018	Food	â‚¹822.0	4	68	Frozen Foods	
10	########	OUT019	Food	â‚¹824.6	4	64	chocolates	
11	########	OUT019	Food	â‚¹824.6	4	60	chocolates	
12	########	OUT019	Food	â‚¹824.6	4	56	chocolates	
13	########	OUT019	Food	â‚¹925.6	4	52	chocolates	
14	########	OUT019	Food	â‚¹1,314.7	4	48	Meat	
15	########	OUT019	Food	â‚¹1,314.7	4	44	Meat	
16								
17								
18								

Fig 9: Exported Sales Data in Excel Format

4.7 Summary of Results

The results of the BigMart Sales Predictor system demonstrate effective and accurate sales forecasting along with real-time inventory management. The system successfully generates predicted sales values, calculates units sold, and updates remaining stock dynamically, as observed in the output dashboard. The sales history table provides a clear record of transactions across different outlets and product categories, enabling easy analysis of trends and performance. Additionally, the export functionality allows users to download structured data for further analysis in Excel, enhancing usability. The inventory status indicators help monitor stock levels and ensure optimal inventory control. Overall, the results confirm that the system provides reliable

predictions, supports efficient stock management, and enables data-driven decision-making in retail operations.

5. CONCLUSION

The project "Sales Forecasting for BigMart using XGBoost: A Machine Learning Approach to Retail Analytics and Inventory Optimization" demonstrates the effectiveness of advanced machine learning techniques in solving real-world retail challenges. By leveraging the power of XGBoost, the system successfully analyzes historical sales data and generates accurate predictions for future demand. The implementation highlights the importance of data preprocessing, feature engineering, and model optimization in improving forecasting performance. Compared to traditional statistical methods, XGBoost provides better accuracy, handles missing values efficiently, and captures complex patterns in large datasets [1]. This model enables retailers to make data-driven decisions, reduce inventory costs, and avoid issues like overstocking and stockouts [2]. Ultimately, the project proves that integrating machine learning into retail analytics significantly enhances operational efficiency and profitability [3].

6. FUTURE SCOPE

Although the current system provides reliable sales predictions, there are several opportunities for further enhancement and expansion. Integration with real-time sales and market data can improve forecasting accuracy and adaptability to changing trends [1]. Advanced machine learning and deep learning models such as Long Short-Term Memory (LSTM) networks can be explored for time-series forecasting to capture sequential patterns more effectively [2]. Automated hyperparameter tuning techniques such as grid search and Bayesian optimization can further improve model performance [3]. The system can also be deployed as a web application using suitable frameworks to provide an interactive dashboard for business users [4]. In addition, integrating the forecasting model with inventory management systems can enable automated stock replenishment and smarter supply chain decisions [5]. Expanding the system to support multi-store and multi-region analysis can improve scalability and business applicability [6]. Finally, incorporating external factors such as weather conditions, promotional campaigns, holidays, and economic indicators can significantly improve prediction accuracy and make the model more robust [7].

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] Kaggle BigMart Sales Dataset, https://www.kaggle.com

[2] Box, G.E.P., Jenkins, G.M., "Time Series Analysis: Forecasting and Control," Wiley, 1976.

[3] Chen, T., Guestrin, C., "XGBoost: A Scalable Tree Boosting System," KDD, 2016.

[4] Fildes, R., et al., "Retail Forecasting: Research and Practice," International Journal of Forecasting, 2019.

[5] Hastie, T., Tibshirani, R., Friedman, J., "The Elements of Statistical Learning," Springer, 2009.

[6] Agrawal, R., et al., "Sales Prediction Using Machine Learning," IEEE, 2021.

[7] Montgomery, D.C., "Introduction to Statistical Quality Control," Wiley, 2009.

[8] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," Proceedings of the 22nd ACM SIGKDD International Conference, 2016.

[9] J. Brownlee, Machine Learning Mastery with Python, Machine Learning Mastery, 2017.

[10] A. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly Media, 2019.

[11] J. Brownlee, Data Preparation for Machine Learning, Machine LearningMastery,2020.

[12] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[13] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016.

[14] A. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow,O'ReillyMedia,2019.

[15] D. Simchi-Levi, X. Chen, and J. Bramel, The Logic of Logistics: Theory, Algorithms, and Applications for Logistics Management, Springer, 2014.

[16] R. Hyndman and G. Athanasopoulos, Forecasting: Principles and Practice, OTexts, 2021.