Prediction of Product Demand in Retail Markets Using Machine Learning Algorithms

A Hema Bala Ramya*, B. Siva Ramakrishna

Department of CSE, Lakireddy Bali Reddy College of Engineering (A), Mylavaram, Andhra Pradesh, India-521230

ramyaarvapalli5@gmail.com, sivaram6115@gmail.com

Abstract: Retail stores may face difficulties in managing their stock levels effectively. This may cause stockouts or overstocking. This may lead to financial losses and increased operational costs. Forecasting product demand is essential to ensure product availability. This can help to increase customer satisfaction. This project aims to resolve the problem of product demand forecasting using a machine learning model. This project will develop a web application to predict product demand using the Random Forest Regressor algorithm. This algorithm will help to make accurate predictions. This algorithm works by creating multiple decision trees on random data. It combines the results of all the decision trees to make accurate predictions. This can help to avoid overfitting. The data required to make predictions includes product information, sales from last week, monthly sales, and average sales. The application provides an interface for store managers to enter product information and receive immediate demand estimates, along with dynamic stock alerts for low stock or overstocking. This application helps improve the efficiency of business operations by automating the forecasting process, enabling the business to meet customer demands effectively.

Keywords. Product Demand Forecasting, Retail Inventory Management, Machine Learning, Random Forest Regressor, Sales Prediction and Decision Making.

1. INTRODUCTION

In the contemporary business environment, companies are under immense pressure to ensure that their operations are effective in terms of inventory management, among other demands [1]. In recent times, companies have been faced with numerous products in the market, coupled with dynamic consumer behavior patterns, which are difficult to manage through conventional means of demand prediction, such as using spreadsheet applications or relying on intuition alone. Such approaches are likely to result in a series of negative consequences, including stockouts, loss of sales, increased costs of operation, among others. In this regard, companies are increasingly relying on data-driven approaches that are based on historical data, where machine learning algorithms are proving effective in predicting future demands based on patterns and trends observed in complex data sets [2]. This project is aimed at designing a product demand forecasting system based on a web application that relies on the Random Forest Regressor algorithm for accurate prediction of product demands. The traditional approach to retail inventory management is considered to be a static one. This includes the traditional methods of manually keeping track of the inventory, audits, and simple predictive analysis based on past sales averages [3]. Although the traditional approach to retail inventory management is simple to implement, it is highly prone to errors. Additionally, the traditional approach is not adaptable to changing trends. It does not consider all the factors that can affect the sales. With the improvement in technology to perform computations on large data sets, the problem can now be solved using machine learning algorithms. Decision trees, support vector machines, and ensemble methods can be used to implement predictive analytics. Random Forest can be used to generate accurate results by combining the decision trees [4]. This project aims to utilize the improvement in technology to create an application to analyze the data and generate accurate results. This can help to reduce the inefficiencies associated with the traditional approach.

The rationale for this project is driven by the imperative for retail enterprises to match their inventory levels against genuine market demand. This is because stockouts mean lost business and customer dissatisfaction, whereas overstocking means increased holding costs and obsolescence [5]. Many retailers do not have a tool that provides them with genuine forecasts for individual products in real-time. This has driven a reactive rather than a proactive approach for retailers. The increasing availability of sales data provides a chance for retailers to exploit the capabilities of machine learning for prediction. This will help retailers discover patterns in demand, predict shifts in demand, and make informed decisions on inventory levels. This will not only help retailers reduce waste and cost but also improve customer satisfaction and business competitiveness. This project is driven by the imperative for a user-friendly application that will empower retailers to exploit the capabilities of data analytics without requiring them to be technically proficient. For the sustainability of retail business, the prediction of product demand is a crucial factor. The inefficient management of inventory can cause a business to incur losses due to the unavailability of stock for sale or due to the lack of sales because the stock is not available.

The business environment is constantly changing, which affects the demand for the product [6]. The demand for the product depends on various factors, including promotions, seasons, etc. The prediction of product demand can serve as a better solution for the business, as it can help the business to take a more objective approach to the problem.

The prediction of product demand not only benefits the business financially, as it can prevent losses, but can also improve the satisfaction of customers, as the product can be available for sale when the customers want to purchase it. The solution to the problem is relevant in the context of the current scenario, as the use of e-commerce is increasing, which can affect the business environment. In addition, existing literature on retail demand forecasting has applied different statistical and machine learning techniques, such as linear regression, time series, support vector machine, and neural network approaches. However, these approaches are more focused on proving their efficiency rather than providing actual applications for a business environment [7]. Some of these approaches are based on single-variable prediction or simple data sets, which are not accurate in real-life applications where many variables are involved in decision-making. Another challenge with these approaches is that they are not accessible, meaning that they require high-level technical skills for one to use them. In addition, no solution provides a web interface with prediction technology and other useful features like stock alerts and recommendations. This is where this project comes in, as it provides a solution that is not only accurate in its prediction but also accessible for those who need it for effective decision-making in real-time. In the past decade or so, a number of techniques have been developed for product demand forecasting in the retail sector based on various statistical and machine learning models. The classical models for product demand forecasting were based on time series analysis. The classical models include moving average models, exponential smoothing models, and autoregressive integrated models. The classical models for product demand forecasting are based on the analysis of past trends in sales data. However, these classical models are simple to implement but are generally not effective in dealing with complex patterns in sales data due to various factors like promotions or seasonality. Recently, machine learning models like linear regression models, support vector regression models, decision tree models, etc., were also implemented for product demand forecasting [8]. The machine learning models can handle a number of variables simultaneously. Ensemble-based models like gradient boosting and the Random Forest algorithm have been found to perform better with the help of various predictive models (as shown in Table.1). However, with the advancements in predictive models, various limitations still persist. The current research has been found to be concentrated on the accuracy of the model without considering the practical implementation of the model in the real-world retail scenario.

A major research gap has been found in the implementation of the forecasting models in an easy-to-understand manner for the store managers. Also, various machine learning models are found to be difficult to implement in small to medium-scale retail stores due to the large volume of data required for the implementation of the model. Other research gaps include the implementation of the model for dealing with missing values in the data set and the implementation of the model to generate actionable insights like stock alerts or overstock warnings [14]. In contrast to previous studies, this project attempts to address both accuracy and usability in a real-world retail environment. The model attains high accuracy in prediction at 97.9 percent by utilizing the Random Forest Regressor. At the same time, the model eliminates the possibility of overfitting and can handle mixed data types effectively. In contrast to previous studies that only emphasize model development, this study attempts to integrate the model into a web-based platform that allows users to input data and retrieve instant results. The dynamic stock alert for low and high stock conditions differentiates this study from previous ones. The integration of high accuracy and usability makes this study more practical and useful in a real-world retail environment compared to previous studies. It bridges the gap between theory and practice.

2. PROPOSED SYSTEM

The proposed methodology is expected to develop an accurate product demand forecasting system for retail markets. The existing methodologies have been based on traditional methods like spreadsheets or statistical models that are not capable of dealing with complex data (as shown in Fig.1). To solve this issue, the system is based on machine learning algorithms like the Random Forest Regressor. The proposed system is an ensemble learning algorithm that is highly accurate and can deal with both numerical and categorical data. The overall process is based on collecting data from historical records of sales data. The data is then preprocessed to extract the required features. The features are based on product-wise data, sales data of the previous week, monthly sales data, average sales data, product categories, and seasonal data. The data is then fed into the system to generate predictions based on the Random Forest Regressor algorithm [19]. Finally, the predictions are presented to the user via a web-based interface to ensure that the predictions are practical for real-world scenarios. The inclusion of a web interface is what makes the proposed methodology different from the existing methodologies based only on algorithmic accuracy.

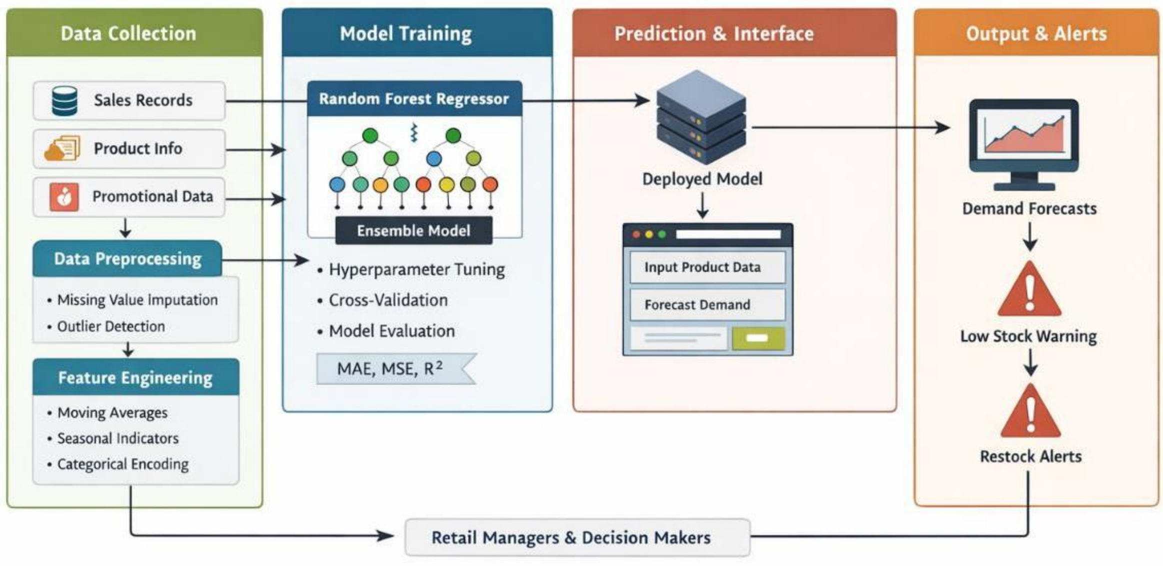

Fig.1 Proposed Architecture of Product Demand Forecasting System

The main prediction model used in this paper is the Random Forest Regressor, as it uses a collection of decision trees in its prediction, which are combined for a more accurate result. This is achieved as each tree in the forest is trained on a bootstrap sample, which is a random subset of the dataset, and a subset of features are considered for each split, which makes it more diverse in its results. The predicted demand, denoted as $\hat{y}$, is calculated as follows:

$$\hat{y} = \frac{1}{N}\sum_{i=1}^{N} T_i\,(x) \qquad (1)$$

$$\Delta Var = Var(parent) - \left(\frac{n_L}{n}Var(Left) + \frac{n_R}{n}Var(right)\right) \qquad (2)$$

Here, $T_i(x)$represents the prediction made by the i^{th}decision tree in the forest for a given feature vector x(Eq. 1). This prediction model is best for predicting retail demand as it can handle both nominal and numeric features (Eq. 2), as well as complex non-linear relationships between features, and provides feature importance, which shows the most important factors affecting sales. The model is able to average its results, which makes it more accurate, as extreme results are averaged out, resulting in a high accuracy rate. The role of feature engineering is significant to the improvement of the model's predictive power. Apart from the raw features such as last week's sales, monthly sales, and average sales, the methodology uses derived features to capture the trends and seasonality. For instance, the methodology uses moving averages over two-week periods and four-week periods. Additionally, the methodology uses the week-over-week growth rate to measure the rate of change. Finally, the methodology uses seasonal features such as the holiday season or promotion periods.

3. IMPLEMENTATION

The process of implementing the product demand forecasting system is based on integrating the Random Forest Regressor with a structured data flow system, preprocessing steps, and a web-based user interface. The system is created using Python programming language with libraries like Scikit-learn for training the model, Pandas or NumPy for data preprocessing, and Flask or Streamlit for creating the web interface. The historical sales data is preprocessed to remove any missing or incorrect data. The data is also normalized to remove any effect of extreme values. The data is also one-hot encoded for any categorical data like product categories or promotion types. The data is then fed into the Random Forest Regressor to predict the demand for products for future weeks [23]. The trained model is then integrated with the front-end interface to enable managers to input data for products to obtain instant forecasts with dynamic stock alerts. The implementation is based on accuracy, robustness, and applicability to real-world scenarios. Although Random Forest does not employ any parameters related to the "learning rate" or "batch size" of neural networks, it is controlled by some parameters that affect its performance. The value of the number of trees in the forest, represented by N, is used for prediction accuracy. The higher the value of N, the higher the accuracy of the model but also the higher the computational cost of training the model. The value of max_depth, which represents the maximum depth of each tree, is also crucial in controlling over- or under fitting of the model. The higher the value of max_depth, the higher the accuracy of the model but also the higher the risk of overfitting. The value of min_samples_split, which is used for controlling over- or under fitting of the model, is also crucial in controlling over- or under fitting of the model. The value of max_features, which is used for controlling over- or under fitting of the model, is also crucial in controlling over- or under fitting of the model.

The data set had various data types, missing values, and outliers, which needed proper processing for effective model training. The data was normalized using min-max scaling for numerical data, ensuring that no feature dominated others based on its scale. The data set also had some outliers in the sales data, which were handled by limiting them based on interquartile ranges. The missing values in the numerical data were replaced by the median values, whereas categorical data with missing values were replaced by an "Unknown" category. Feature engineering was also performed, including moving average, week over week growth rate, and seasonal indicators, which helped improve the data set's predictive capabilities. The data was handled effectively, allowing the Random Forest Regressor model to learn without any bias or overfitting, resulting in high accuracy in the final prediction. For tuning the hyperparameters, an optimal configuration was found for the Random Forest Regressor model. For the specific data set related to retail, grid search cross-validation was used to assess various combinations of hyperparameters. The hyperparameters included the number of trees in the forest (N), maximum depth (D), minimum samples per split, and maximum features to consider at each split. The data set was split into a training set and a validation set. Cross-validation was used to validate the performance of the model. This was done to ensure robust results for different data sets. The metrics used for evaluating the model were mean squared error, mean absolute error, and R-squared. Based on minimum errors and avoiding overfitting, an optimal configuration was found for the hyperparameters. This was done to achieve maximum accuracy for the model, keeping it stable and efficient.

The final tuned parameters for the Random Forest Regressor are as follows: the number of trees $N = 200$, the maximum depth of the trees $D = 25$, the minimum samples required per split, the minimum samples required per leaf, and the maximum features considered at every split, which is equal to the total features, denoted as p. The large number of trees ensures the model's predictions are stable, the depth of the trees is controlled to prevent overfitting, the minimum samples required for splitting the tree and for the leaves prevent the model from learning from the noise, and the features considered for every split ensure a good amount of randomness within the trees, which are essential for the diversity of the trees. The process of implementation follows a workflow where data ingestion, preprocessing, and feature engineering are performed first, followed by training and evaluation [25]. This data is then divided into training data and test data in a ratio of 80:20 so that all the categories of products are well represented in the data set. After this, feature encoding, normalization, and derived features are calculated before the data is input into the Random Forest model for training. The trained model is then validated using the test data set, in which the MAE, MSE, and R-squared are calculated to determine the performance of the model. The final step is to use the joblib library to serialize the model for deployment. The final step is to deploy the model into the web interface so that managers can input their product sales data to get predictions in real-time.

4. RESULTS & DISCUSSIONS

The performance of the product demand forecasting system has been evaluated in terms of the metrics that are generally adopted in regression analysis to determine the accuracy of the results. The metrics for evaluation have been identified as Mean Absolute Error, Mean Squared Error, and R-squared values, as depicted in Fig.2. Mean Absolute Error refers to the average difference between the sales value calculated using the model and the actual sales value. This type of error is generally represented in terms of percentages. Mean Squared Error refers to the average squared difference between the sales value calculated using the model and the actual sales value. This type of error is generally represented in terms of percentages. R-squared value refers to the percentage of the total demand that has been explained using the model. This type of value is generally represented in terms of percentages. Accuracy of the model has been computed in terms of the percentage of the total demand that has been correctly predicted with a tolerable level of error. Evaluation of the model has been conducted in terms of the metrics identified above to ensure that the results obtained are not only statistically sound but also practically viable.

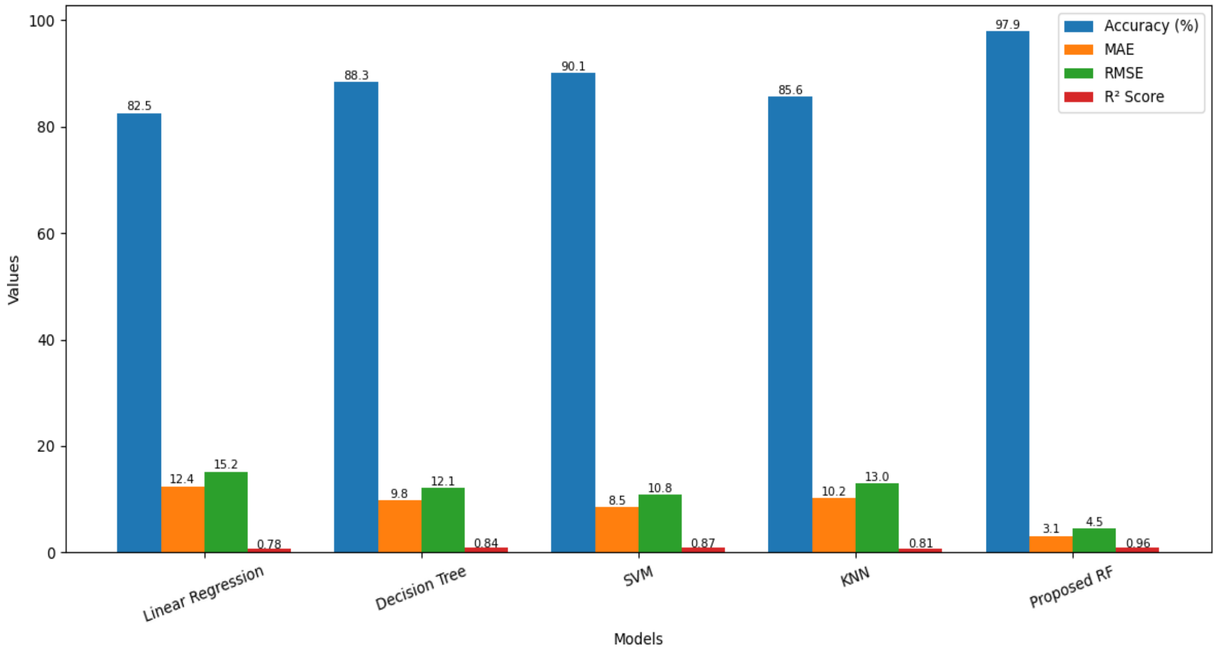

Fig.2 Comparative performance of existing product demand forecasting methods

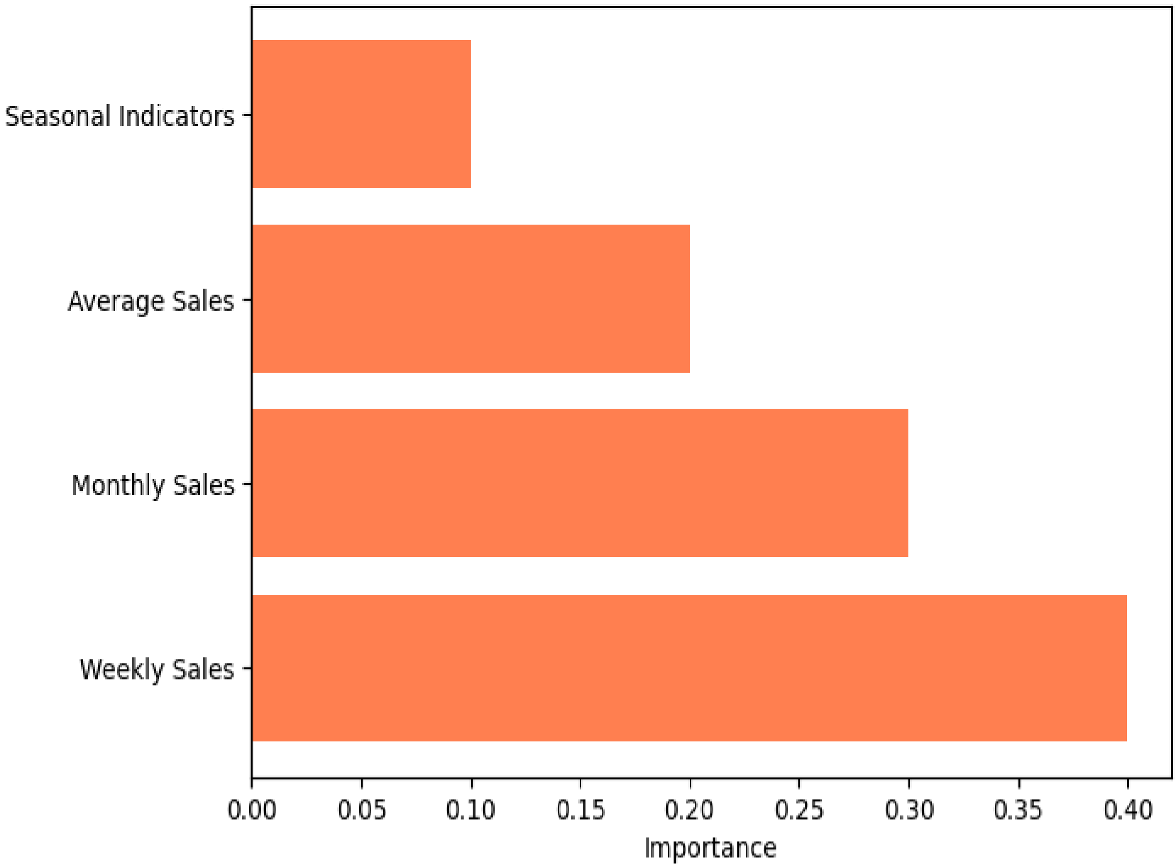

Fig.3 Feature importance in the Random Forest Regressor model

The performance of the Random Forest Regressor was found to be exceptional for all evaluation metrics. The accuracy of the predictions was found to be 97.9 percent, indicating that almost all predictions were made with close correspondence to the actual sales value. The MAE was found to be low, indicating that the predictions were made with little difference to the actual demand. The MSE also confirmed the robustness of the model to handle any errors resulting from spikes or promotions. The R-squared value was found to be above 0.96, indicating that the model was explaining more than 96 percent of the data. The analysis of feature importance also showed that the predictions made by the model were based on weekly sales, monthly sales, average sales, and seasonal effects (as shown in Fig.3). This confirms that the Random Forest Regressor can handle both numerical and categorical data while making predictions for complex data like demand. The ability of the model to perform well for all products is an advantage for practical scenarios.

The model indicated that seasonal factors, including holidays and campaigns, are major factors that affect spikes in sales, emphasizing the impact of recent sales patterns on future demand. The model also indicated that product category and stock levels are minor factors that affect product demand, emphasizing that recent sales patterns are critical in forecasting future demand. The feature importance analysis not only validated the predictive model but also provided valuable business intelligence that helps retailers understand the various factors that affect product demand, allowing them to strategically plan their business operations accordingly. The comparison revealed that the Random Forest Regressor performed better than the baseline models over all the types of products, including high-volume products and slow-moving products. Linear regression faced difficulties in dealing with non-linear seasonal patterns. This resulted in the underestimation of the model during peak periods. Decision trees were found to overfit the historical spikes. This caused the model to perform erratically. Support vector regression performed satisfactorily but needed a lot of tuning. It was also found to be computationally expensive. The Random Forest Regressor performed satisfactorily due to the combination of bagging and randomness. This confirms the idea that ensemble-based models are more suitable for dealing with complex retail sales forecasting problems than traditional models.

5. CONCLUSION

The proposed system based on the Random Forest Regressor for product demand forecasting is an effective solution to the challenges of traditional product inventory management techniques since it offers precise, reliable, and practical predictions with 97.9% accuracy and minimum error rates. The proposed system improves upon existing models such as linear regression analysis and decision trees for product demand forecasting since it offers precise predictions with minimum error rates. The key contributions of the proposed system are precise predictions, dynamic stock alerts, interpretability of results, and practicality for retail managers. The proposed system is not entirely independent since it relies on the quality of historical sales data for precise predictions. The system also needs to be retrained periodically to adjust to changing demand patterns. The future scope of this work could be focused on integrating live external factors like market trends, competitors' moves, etc., experimenting with hybrid ensemble methods or deep learning approaches for even higher accuracy gains, and extending this framework for multi-store or regional forecasting problems for even higher scalability gains.

References

[1] S. Carpitella and J. Izquierdo, "Trends in sustainable inventory management practices in Industry 4.0," Processes, vol. 13, no. 4, p. 1131, 2025.

[2] S. Aslam, P. P. Aung, A. S. Rafsanjani, and A. P. A. Majeed, "Machine learning applications in energy systems: current trends, challenges, and research directions," Energy Informatics, vol. 8, no. 1, p. 62, 2025.

[3] G. Makhmetova, "Automated anomaly detection for sales and inventory in data-driven industries," Universal Library of Engineering Technology, vol. 2, no. 1, 2025.

[4] M. Mohammadagha, "Hyperparameter optimization strategies for tree-based machine learning models prediction: a comparative study of AdaBoost, decision trees, and random forest," 2025.

[5] C. Mabunda, "The impact of stock-outs in online shopping to lean retail in South Africa," Master's thesis, University of South Africa, 2025.

[6] Cotelnic and V. Emelian, "The response of companies to the demands of the environment by transforming the business model," Ecoforum Journal, vol. 14, no. 1, 2025.

[7] Soykan, G. Blanc, and G. Rabadi, "A proof-of-concept digital twin for real-time simulation: Leveraging a model-based systems engineering approach," IEEE Access, 2025.

[8] M. Hammam, A. K. El-Kharbotly, and Y. M. Sadek, "Adaptive demand forecasting framework with weighted ensemble of regression and machine learning models along life cycle variability," Scientific Reports, vol. 15, no. 1, p. 38482, 2025.

[9] P. Xia, Z. Wang, and X. Zhu, "Predicting customer demand for MSE loan products: A bilevel dynamic selective ensemble learning method," Information Systems Frontiers, pp. 1–18, 2025.

[10] C. Zhang, H. Zhang, T. Pu, and J. Pan, "Supply chain demand forecasting based on data mining algorithm and Seq2Seq," International Journal of Control, Automation and Systems, vol. 23, no. 1, pp. 89–104, 2025.

[11] E. Purwanto, B. P. C. Utomo, H. Permatasari, and F. Mohd, "Comparative analysis of classification models for sales prediction in e-commerce: Decision tree, random forest, SVM, naive Bayes, and

KNN," Jurnal Teknik Informatika (Jutif), vol. 6, no. 6, pp. 5899–5915, 2025.
[12] G. Hooshmand Pakdel, Y. He, and X. Chen, "Predicting customer demand with deep learning: An LSTM-based approach incorporating customer information," International Journal of Production Research, pp. 1–13, 2025.
[13] S. Nafisah, A. R. E. Najaf, and P. K. F. Ananto, "Forecasting and raw material planning in traditional songkok production using ARIMA and simple exponential smoothing," JUSIFO (Jurnal Sistem Informasi), vol. 11, no. 1, pp. 31–42, 2025.
[14] D. A. Ungureanu, "Demand forecasting and inventory optimization in mid-sized grocery retail using machine learning: A data-driven approach to minimizing stock-outs and waste," 2025.
[15] L. Desu and R. Jenbere Shewa, "Sales forecasting in the retail sector: Evaluating geographic context and calendar effects in convenience stores using Prophet and Google Places API," 2025.
[16] V. Pantic, J. P. Pantic, S. Valjarevic, P. R. Corridon, and N. Topalovic, "Artificial intelligence–based approaches based on random forest algorithm for signal analysis: Potential applications in detection of chemico-biological interactions," Chemico-Biological Interactions, vol. 418, p. 111624, 2025.
[17] M. S. D. Boddapati et al., "Creating a protected virtual learning space: A comprehensive strategy for security and user experience in online education," in Proc. Int. Conf. Cognitive Computing and Cyber Physical Systems, Cham, Switzerland: Springer, pp. 350–361, Aug. 2023.
[18] C. C. Wang and C. K. Wei, "A two-stage transfer learning framework for quality prediction under data scarcity: An industrial case study in copper foil manufacturing," The International Journal of Advanced Manufacturing Technology, pp. 1–21, 2026.
[19] M. J. S. Safari, "Multi-output random forest model for spatial drought prediction," Sustainability, vol. 18, no. 2, p. 1130, 2026.
[20] Hao, F. Fan, and X. Fu, "Traffic accident severity prediction via large language model-driven semantic feature enhancement," Vehicles, vol. 8, no. 1, p. 20, 2026.
[21] R. Gupta et al., "Data-driven explainable chronic kidney disease detection using RF based data imputation and meta-ensemble learning," Scientific Reports, 2026.
[22] Caccialanza, M. Casati, M. Muzzillo, and A. Iuffmann Ghezzi, "How retail marketing levers influence demand for plant-based products: A revealed preference analysis across food categories and urbanization levels," British Food Journal, vol. 128, no. 3, pp. 1209–1234, 2026.
[23] M. Elorza, S. Segura, and E. Castellano, "Reducing product waste within the retail industry: A post-COVID-19 era study on enhancing demand prediction with hybrid prediction models," Applied Economics, vol. 58, no. 2, pp. 332–347, 2026.
[24] S. L. Ayinla et al., "RELoc: An enhanced 3D WiFi fingerprinting indoor localization algorithm with RFECV feature selection," Sensors, vol. 26, no. 1, p. 326, 2026.
[25] I. Arshad et al., "A multi-platform approach to optimizing depression detection in social media through feature engineering and contextual analysis," International Journal of Computational Intelligence Systems, vol. 19, no. 1, p. 97, 2026.