Journal of
Supply Chain Management Science

TU Delft OPEN Publishing

Research Article

AI-driven raw material demand forecasting: Towards project management practices

Abdelrahman Mohamed Fouad [1,*], Hossam Wefki [1], Hanan Kouta [2]

[1] Civil Engineering Dept., Faculty of Engineering, Port Said University, Egypt;
[2] Production Engineering and Mechanical Design Department, Faculty of Engineering, Port Said University, Egypt

*Corresponding author: abdelrahman.fouad@eng.psu.edu.eg

Abstract: Accurate forecasting of raw material demand is critical for effective decision-making in project management, particularly in the construction sector. Demand fluctuations in this industry can severely disrupt workflows and cause project delays. Reliable demand predictions help maintain operational stability and reduce risks, especially under uncertain conditions. In such environments, maintaining appropriate Target Stock Levels (TSL) and setting effective Reorder Points (RP) are essential to ensure project continuity and customer satisfaction. This study investigates the potential of Artificial Intelligence (AI) to enhance demand forecasting accuracy within service-based supply chains. Five forecasting models were evaluated: four machine learning approaches—Extreme Gradient Boosting (XGBoost), Long Short-Term Memory (LSTM), Random Forest (RF), and Gaussian Regression (GR)—and one traditional statistical method, the Autoregressive Integrated Moving Average (ARIMA) model. The models were trained and tested using historical raw material consumption data collected from a construction project over a four-year period (2019–2023). The results show that the Machine Learning Models significantly outperformed the ARIMA model in terms of predictive accuracy. The coefficients of determination (R^2) were 0.93 for LSTM, 0.91 for XGBoost, 0.89 for GR, and 0.88 for RF, compared to 0.75 for ARIMA. Among all models, LSTM achieved the highest forecasting accuracy and the lowest deviation on the test dataset. Its implementation for the 2024 planning horizon led to substantial inventory optimization, reducing overstock volumes by 66.5%. This improvement translated into significant cost savings and enhanced the overall efficiency of material management and decision-making processes.

Keywords: Demand forecasting; Artificial intelligence; Time series prediction; Construction supply chain seasonality

Publishing history: Submitted: 22 April 2025; Revised: 21 May 2025, 17 June 2025, 06 July 2025; Accepted: 07 July 2025; Published: 13 October 2025

Cite as: Fouad, A.M., Wefki, H. & Kouta, H. (2025) AI-Driven Raw Material Demand Forecasting: Towards Project Management Practices. *Journal of Supply Chain Management Science*, *6*(1-2). https://doi.org/10.59490/jscms.2025.8136

ISSN: 2451-9901
Vol. 6(1-2), 2025
DOI: 10.59490/jscms.2025.8136

1 Introduction

In today's globalized market, Supply Chain Management (SCM) plays a pivotal role in enabling efficient resource utilization and operational continuity. This is particularly critical in the construction sector, where dynamic demand, project-based operations, and dependence on multi-tiered suppliers expose the supply chain to substantial uncertainty and risk (Christopher and Peck, 2004; Dey et al., 2015). Delays or overstocking can significantly impact project timelines and inflate costs.

The COVID-19 pandemic further exposed systemic vulnerabilities in global supply chains, including those in the construction industry. Lockdowns, transportation restrictions, and raw material shortages led to widespread disruptions, highlighting the urgent need for resilient and accurate forecasting models to maintain continuity and buffer against uncertainty (Ivanov and Das, 2020; Golan et al., 2020).

Forecasting is a core function in SCM, influencing procurement, inventory control, production scheduling, and customer satisfaction (Rahman et al., 2017). When accurate, forecasting enables firms to align supply with anticipated demand, minimize stockouts or overstocking, and reduce overall operational costs (Chopra and Meindl, 2016). However, the complexity of forecasting has increased markedly over the past two decades, driven by globalized sourcing, demand volatility, and the exponential growth of data generated through ERP systems, IoT, and digital platforms (Manyika et al., 2011; Chen et al., 2014).

Traditional statistical methods such as (ARIMA) model have been widely employed for time series forecasting, primarily due to their interpretability and strength in modelling linear relationships (Box et al., 2015). However, these models struggle to capture nonlinearities and high-dimensional dynamics prevalent in modern industrial data. As emphasized by Hyndman and Athanasopoulos (2021), ARIMA's limitations have prompted a shift toward AI-based and machine learning (ML) approaches capable of learning from complex patterns and irregular time series (Makridakis et al., 2018).

Advanced ML techniques such as (LSTM) networks, (XGBoost), (RF), (GR) have shown substantial promise in demand forecasting applications. These methods offer strong generalization capabilities and robustness to noise, enabling them to outperform traditional models in many industrial and retail forecasting tasks (Lim and Zohren, 2020; Zhang, 2002).

While Bandara, Bergmeir, and Smyl (2020) and Liu, Zhang, and Zhang (2025) propose a general hybrid framework that integrates ARIMA with deep learning models to enhance forecasting accuracy, this study uniquely compares five forecasting models using construction-specific raw material demand data. This dataset and domain remain underrepresented in prior forecasting research. The aim is to evaluate the relative strengths of these models using real raw material demand data from the construction sector, providing a focused contribution to an area seldom addressed in comparative forecasting literature.

Moreover, construction demand is multifactorial, influenced by seasonality, project phase, procurement lead times, and external economic conditions. As Fildes et al. (2019) argue, forecasting models that integrate such contextual variables demonstrate significantly higher reliability and relevance in practical applications (Brownlee, 2020).

This study compares five forecasting methods—ARIMA, XGBoost, LSTM, Random Forest, and Gaussian Regression—based on actual usage data from a leading Egyptian construction manufacturer. Its primary contributions include: (i) Demonstrating the superior performance of LSTM ($R^2 = 0.93$) compared to other models (XGBoost: 0.91, RF: 0.88, GR: 0.89, ARIMA: 0.75), (ii) Reporting a 66.5% reduction in overstock levels in 2024, indicating significant cost savings and inventory efficiency and (iii) Offering a scalable AI-based solution for data-driven decision-making in raw material procurement.

The paper is structured as follows: Section 2 reviews relevant literature; Section 3 outlines the methodology and data; Section 4 presents the case study; Section 5 discusses the results; and Section 6 concludes with key findings and implications.

2 Literature Review

This section critically reviews key developments in demand forecasting within (SCM), focusing on time series models and machine learning (ML) approaches used for raw material demand forecasting. The literature is grouped into four thematic areas: (1) classical time series methods, (2) machine learning and deep learning models, (3) evaluation metrics, and (4) external factors influencing demand forecasting.

The ARIMA model has long been a foundational method for time series forecasting due to its capacity to model temporal dependencies and non-stationary data. In the context of SCM, especially in the construction sector (Salem et al., 2023). ARIMA has been used to forecast demand for materials such as cement and steel, where lead times and historical usage patterns are essential for planning (Hyndman and Athanasopoulos, 2021; Makridakis et al., 2018). ARIMA's linear structure allows for interpretable forecasts, but its reliance on stationarity and limited ability to capture complex nonlinear dynamics poses challenges in volatile or multi-factorial environments (Fildes et al., 2019). While ARIMA remains a benchmark model, its performance often degrades in scenarios with high volatility or irregular consumption patterns, which are common in project-based industries like construction (Brahami et al., 2021; Brownlee, 2020). These limitations have prompted a shift toward more adaptive, data-driven models.

Machine learning models such as Random Forest and XGBoost have been widely adopted in recent forecasting research due to their flexibility in modeling nonlinear relationships and handling large feature spaces (Fattah et al., 2018). XGBoost has shown superior performance over traditional models in retail demand forecasting, suggesting potential applicability to raw material forecasting where demand is influenced by multiple dynamic factors (Lim and Zohren, 2020)

Deep learning models, particularly LSTM networks, have also gained traction for their ability to model long-range dependencies in sequential data (Moutacalli et al., 2014). LSTM is particularly well-suited to construction supply chains, where projects involve multi-phase schedules and seasonally fluctuating material needs (Zhang, 2002; Bandara et al., 2020).

Liu, Zhang, and Zhang (2025) introduce a hybrid ARIMA-deep learning framework and Hyndman and Athanasopoulos (2021) proposed hybrid models that combine ARIMA with Machine Learning or (AI)-based models to leverage both statistical rigor and non-linear learning capabilities. While these hybrid approaches demonstrate potential, they often involve complex tuning procedures and significant computational demands, and their effectiveness tends to vary across different application sectors. Although the present study does not implement hybrid models, it draws insights from these prior contributions to benchmark the performance of standalone forecasting models (Noh et al., 2021).

The selection of performance metrics is critical to model assessment. Widely used metrics include Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). Taylor (2020) argues for a multi-metric evaluation framework to reflect various dimensions of model performance, such as robustness to outliers and interpretability across time horizons. These metrics are essential in construction SCM, where underestimating demand can delay projects, and overestimating can tie up working capital.

Fildes et al. (2019) and Brownlee (2020) emphasize that specific evaluation metrics—especially MAPE—may exhibit excessive sensitivity to low demand values, potentially distorting forecast accuracy assessments for low-volume, high-value materials. To mitigate this issue and enhance the reliability of forecasting in practical applications, Ali et al. (2011) advocate for the use of multiple complementary metrics during model evaluation.

Recent literature has emphasized the importance of contextual and external variables such as economic cycles, seasonal trends, supplier reliability, and logistical constraints on forecasting outcomes. Zhang and Hong (2022) highlight the improved accuracy when models integrate such factors, especially in sectors where historical trends do not solely drive demand. In construction, weather conditions, labor availability, and regulatory delays also introduce variability that standard models may not adequately address.

While broader crises like the COVID-19 pandemic and geopolitical events have underscored the vulnerability of global supply chains, forecasting models must be adaptable to rapid shifts in project schedules and material availability (Abdelsalam et al., 2020).

In summary, classical models like ARIMA offer simplicity and interpretability but often fall short in capturing complex demand dynamics typical in construction. Machine learning and deep learning models provide more accurate forecasts under variable conditions but may lack transparency and require significant data preprocessing. Despite the extensive application of forecasting models in broader supply chain research, Wang et al. (2020) and Lim and Zohren (2020) and Khan (2021) note that direct comparisons between ARIMA, LSTM, (RF), XGBoost, and (GR) remain limited particularly in the context of raw material demand forecasting within the construction sector.

This study addresses this gap by applying and comparing five models ARIMA, LSTM, RF, XGBoost, and GR—on actual construction material demand data. It contributes to the literature by identifying model strengths and limitations under construction-specific demand conditions and establishing a performance benchmark using RMSE, MAE, and MAPE.

3 Methodology and Data Collection

This research employs a practical, domain-specific methodology to forecast raw material demand in the construction sector, focusing on a real-world service supply chain context. The approach combines advanced forecasting techniques such as ARIMA, XGBoost, LSTM, Random Forest, and Gaussian Regression. Each model is calibrated to reflect the dynamic consumption patterns observed in construction projects, where material demand fluctuates due to schedule shifts, labor availability, and weather-related delays.

3.1 Data collection

Data were collected from four consecutive years' procurement logs and inventory records, encompassing monthly consumption values for essential construction materials such as cement, rebar, and concrete blocks. Each entry includes the project ID, delivery date, and site location.

Preprocessing procedures included:

- Missing value imputation using linear interpolation for continuous variables.
- Outlier removal using IQR filtering and visual inspection to correct anomalies such as duplicate or delayed entries.
- Transformation to time series format, indexed by month and project ID.

Feature Engineering:

- Lag Features: Created for lags 1, 2, and 3 months to capture short-term autocorrelation.
- Rolling Window Statistics: 3-month rolling mean and rolling standard deviation were added.
- Seasonal Dummies: Month-of-year encoded as 12 binary dummies to capture seasonality.
- External Regressors: Project phase indicators and rainfall data (as a weather proxy) were included.

3.2 Model performance assessment

The effectiveness of the forecasting models is evaluated using standard statistical performance metrics—Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the coefficient of determination (R^2). These metrics quantify the accuracy of predictions by comparing them against actual demand values.

In addition to numerical metrics, diagnostic visualizations such as residual and Q–Q plots are used to validate model assumptions and detect anomalies. According to Huber and Stuckenschmidt (2020) such tools are vital for identifying areas where model performance may be compromised. Liu et al. (2025) also underscore the importance of visual diagnostics in enhancing the interpretability of forecasting models in complex prediction tasks.

3.3 Hyperparameter optimization

To improve forecasting accuracy, hyperparameters of the applied models are fine-tuned. This study systematically applies cross-validation and grid search techniques to explore the hyperparameter space for each model.

These optimization strategies ensure the models balance between underfitting and overfitting, thus improving generalizability. Wu, Li, Lim, and Li (2023) emphasized the value of hyperparameter tuning in time series models like ARIMA and LSTM, while Yang, Li, and Rasul (2021) demonstrated that careful tuning significantly boosts accuracy in real-world forecasting scenarios.

3.4 Demand prediction models

Based on historical demand data, five distinct forecasting models ARIMA, XGBoost, LSTM, Random Forest, and Gaussian Regression—are trained and tested. These models were selected due to their demonstrated effectiveness in prior research, such as the work of Bandara, Bergmeir, and Smyl (2020) for ARIMA, Chen M. , Mao, and Liu (2014) for XGBoost, Hochreiter and Schmidhuber (1997) for LSTM, and Breiman (2001) for Random Forest.

The comparative framework adopted in this study enables a comprehensive evaluation of each model's forecasting capability within the context of raw material demand. Jha et al. (2021) emphasized integrating machine learning and traditional models in supply chain demand forecasting. Studies by (Fattah et al., 2018) validated the robustness of models like Random Forest and ARIMA, while Jin et al. (2022) recommended hybrid approaches (e.g., ARIMA–LSTM) to handle complex temporal variations as illustrated in Figure 1.

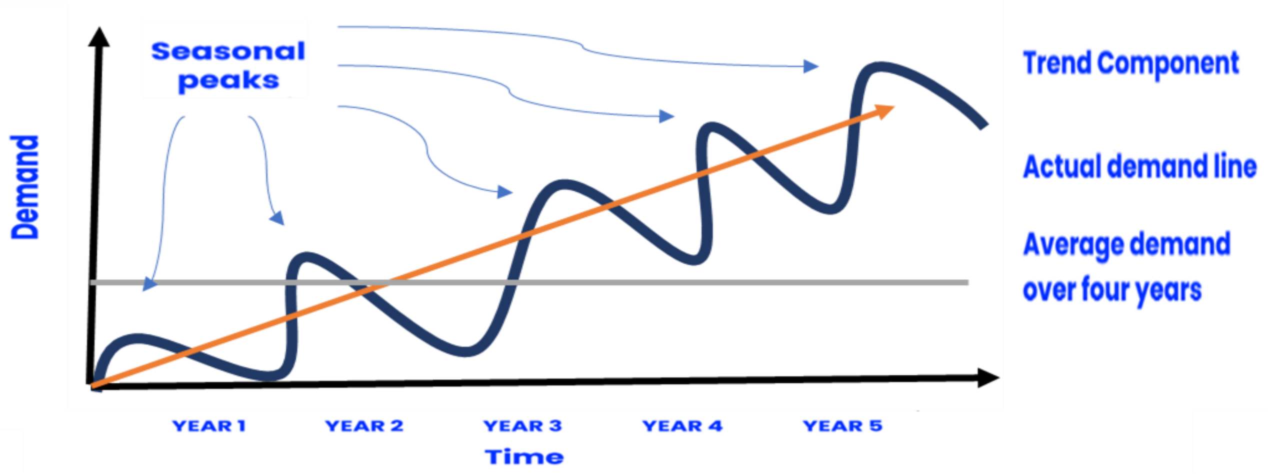

Figure 1: Demand patterns over time

3.5.1 Extreme Gradient Boosting

XGBoost is a high-performance, scalable tree-boosting algorithm widely adopted in prediction tasks. In this study, it forecasts raw material demand by capturing patterns through an ensemble of weak learners optimized with a convex differentiable loss function. Typical loss functions include MAE and MSE, which guide model training through gradient descent.

Objective Function:

$$\text{Obj} = \sum_{i=1}^{n} L(y_i, \hat{y_i}) + \sum_{k=1}^{K} \Omega(f_k) \tag{1}$$

Regularization Term:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} \omega_j^2 \tag{2}$$

In XGBoost, the optimization process relies on a differentiable convex loss function, typically chosen to facilitate effective gradient-based optimization. This function is shaped by error metrics such as (MSE) and (MAE), which assess how well the model's predictions align with actual outcomes, and contribute to refining the model's parameters during training.

The model was developed through a systematic process:

Step 1: Data Preparation
Raw monthly demand data were cleaned and restructured to facilitate supervised learning.

Step 2: Feature Engineering
Time-lag features were created to capture sequential dependencies and seasonal variations.

Step 3: Data Structuring
The dataset was formatted into input-output pairs suitable for XGBoost regression modeling.

Hyperparameters Tuned:

- Learning Rate (0.05): Controls the impact of each tree on the final prediction.
- Max Depth (10): Limits the depth of individual trees to prevent overfitting.
- Number of Estimators (300): Specifies the number of boosting rounds.
- Subsample (0.8): Determines the fraction of observations used per tree.
- Colsample_bytree (0.8): Controls the fraction of features considered at each split.
- Gamma (0.1): Regularization parameter to reduce unnecessary splits.
- Alpha – L1 Regularization (0.5): Encourages sparsity in the model.
- Lambda – L2 Regularization (1.0): Reduces model complexity.

Step 4: Model Tuning
Grid search with cross-validation was applied to identify the optimal hyperparameter configuration.

3.5.2 Long Short-Term Memory

LSTM networks are a type of recurrent neural network (RNN) designed to handle sequential data. Their architecture includes memory cells and gates (input, forget, output) that regulate the flow of information over time. This allows LSTM models to retain long-term dependencies, making them suitable for time series forecasting with seasonal and trend variations.

Forget Gate:

$$f_t = \sigma(W_f * [\,h_{t-1}, x_t] + b_f \tag{3}$$

Input Gate:

$$i_t = \sigma(W_i * [\,h_{t-1}, x_t] + b_i \tag{4}$$

Cell State:

$$\overline{C_t} = W_c * [\,h_{t-1}, x_t] + b_c \tag{5}$$

$$C_t = f_t * C_{t-1} + i_t * C_t \tag{6}$$

Output Gate:

$$O_t = \sigma(W_o * [h_{t-1}, x_t] + b_o \quad (7)$$

Hidden State:

$$h_t = o_t + \tanh(C_t) \quad (8)$$

LSTM networks are specialized for time series forecasting due to their ability to retain long-term dependencies through memory cells and gating mechanisms, as per Figure 2.

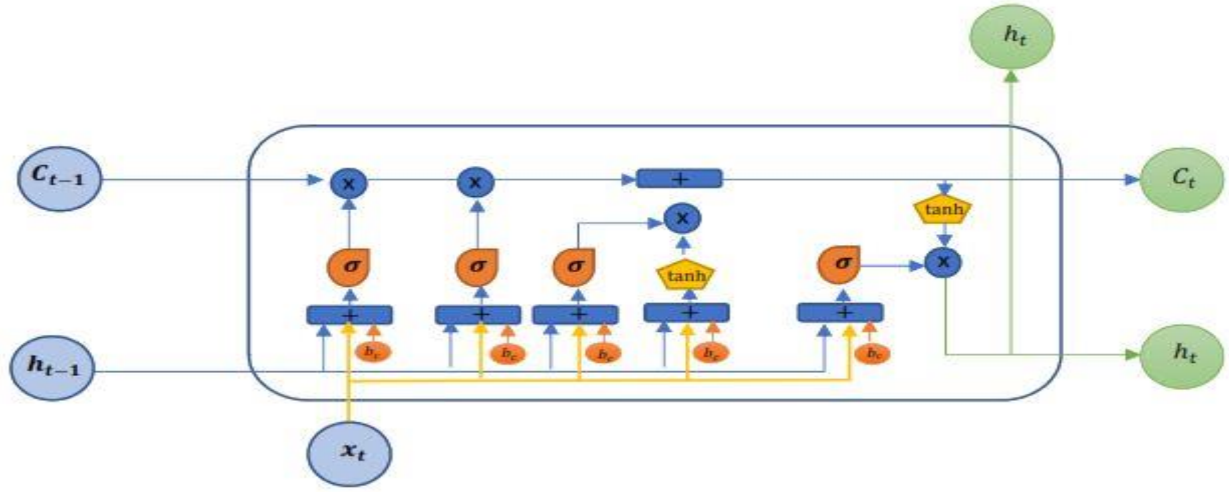

Figure 2: LSTM cell mechanism flow chart

Forget Gate: This gate decides which information from the previous time step should be discarded. It produces a value between 0 and 1, with 0 indicating complete forgetfulness and 1 meaning full retention.

Input Gate: Determines what new data should be incorporated into the cell state. It includes:

- A sigmoid function that selects the data to be updated.
- A tanh function that generates potential values for updating the cell state.

Cell State Update: The forget gate and input gate work together to update the cell state, balancing the retention of past information and adding new data.

Output Gate: It regulates which part of the updated cell state should be output. A sigmoid function filters the data, followed by a tanh function that normalizes the output between -1 and 1.

Using these gates, the LSTM cell effectively learns which information to retain, forget, and output, allowing it to handle complex temporal dependencies in data sequences.

The LSTM forecasting model has different steps. These are:

Step 1: The dataset was structured with look-back windows to form sequences of input-output pairs.

Step 2: Network Architecture

LSTM Model Layer, as depicted in Figure 3, is as:

First Layer: 100 units with full sequence output.
Dropout Layer: Dropout rate of 0.2 to prevent overfitting.
Second Layer: 100 units with full sequence output.
Third Layer: 50 units returning the final output.
Dense Layer: 50 neurons to match the forecast horizon

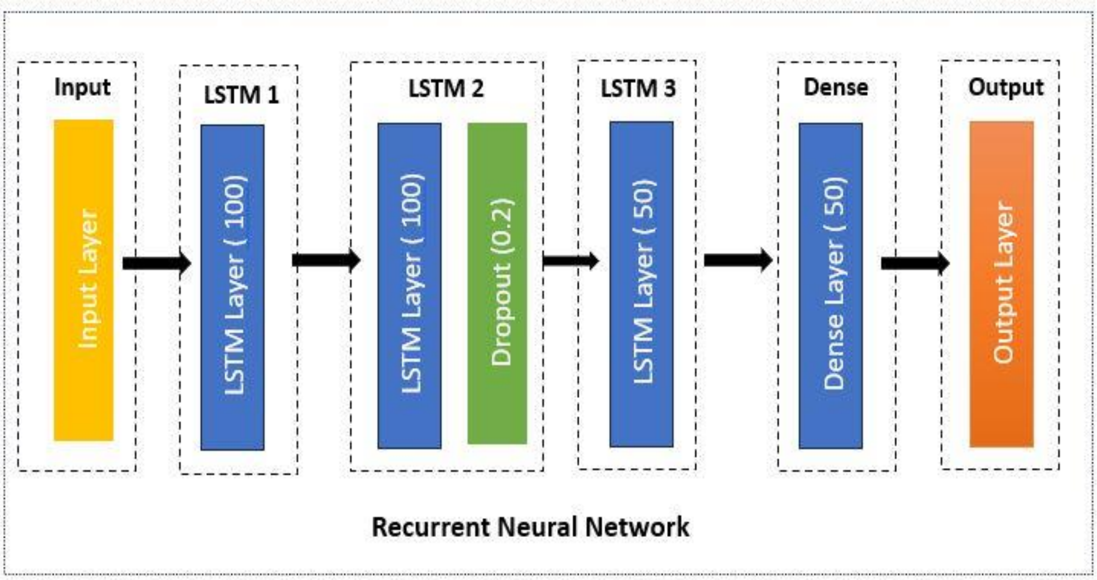

Figure 3: LSTM cell mechanism flow chart

Step 3: Output Formatting
The output sequence was reshaped to match the format of the future time steps being forecasted.
Hyperparameters

- Number of Layers: Depth of the network.
- Number of Units: Neurons per layer.
- Dropout Rate: Prevents overfitting.
- Batch Size: Samples per gradient update.
- Learning Rate: Step size for optimization.
- Epochs: Iterations over the entire dataset.

Step 4: Training and Tuning
The model was trained and validated using time-based cross-validation. Hyperparameters were optimized using grid search.

3.5.3 Random Forest

Random Forest is an ensemble learning technique that constructs multiple decision trees and aggregates their outputs to improve prediction robustness. Each tree is trained on a bootstrap sample with randomly selected features. This randomization reduces overfitting and enhances model generalizability. The final output is typically obtained via averaging (regression) or majority voting (classification). The mathematical formulation for the RF classifier is expressed as follows:

$$\hat{y} = \frac{1}{M}\sum_{m=1}^{M}\hat{y}_m \tag{9}$$

Where $\hat{y}$ is the prediction for regression. The prediction for classification can be expressed as:

$$\hat{y} = mode(\hat{y}_1, \hat{y}_1, \dots, \hat{y}_M) \quad (10)$$

Each decision tree, denoted a $Ti(x)$, is constructed using a randomly selected subset of the training dataset and a distinct subset of input features. The trees are typically expanded to their full depth or until a specific stopping condition is satisfied.

Random Forest is an ensemble learning method based on constructing multiple decision trees and aggregating their outputs.

Step 1: Data Preprocessing
Time series data was cleaned and formatted. Lag features were created to represent prior periods.

Step 2: Feature Engineering
Additional variables, such as rolling averages and month identifiers, were included to improve model generalization.

Step 3: Model Training
Multiple decision trees were trained on bootstrapped samples, and outputs were averaged for prediction.

Hyperparameters

- Number of Trees (200): Total decision trees used in the ensemble.
- Maximum Depth (15): Maximum allowable depth for individual trees.
- Min Samples Split (4): Minimum samples required to split a node.
- Min Samples per Leaf (2): Minimum samples required at each leaf node.
- Bootstrap (True): Enables random sampling with replacement.
- Criterion ("MSE"): Metric used to evaluate splits (mean squared error).

Step 4: Optimization
Grid search and k-fold cross-validation were used to determine optimal model parameters.

3.5.4 Gaussian Regression

GR is a Bayesian, non-parametric approach that estimates the predicted value and the associated uncertainty. The model leverages a kernel (covariance) function, denoted as $k(x, x')$, to capture the similarity between data points. This enables GR to model complex relationships while providing confidence intervals for predictions.

The Gaussian regression can be defined as:

$$f(x) \sim \varsigma p(m(x), k(x, x')) \quad (11)$$

The mean function is given in (12), while the covariance function is expressed in (13).

$$m(x) = E[f(x)] \quad (12)$$

E : Expectation operator.

$$K(x, x') = E[(f(x) - m(x))(f(x') - m(x'))] \quad (13)$$

The prediction equations are:

$$f_* \mid X, y, X_* \sim N(\bar{f}_*, cov(f_*)) \tag{14}$$

$$\bar{f}_* = K(X_*, X)[K(X, X) + \sigma_n^2 I]^{-1} y \tag{15}$$

$$one\ cov(f_*) = K(X_*, X_*) - K(X_*, X)[K(X, X) + \sigma_n^2 I]^{-1} K(X, X_*) \tag{16}$$

Gaussian Regression provides probabilistic predictions and includes confidence intervals around forecasts.

Step 1: Data Preparation
Historical demand was cleaned and missing values were filled out through filled through linear interpolation.

Step 2: Normalization
Input features were scaled to ensure numerical stability during model training.

Step 3: Feature Engineering and Tuning
Temporal features such as lag values and cyclical time representations were added. Model tuning was performed using cross-validation.

Hyperparameters
- Kernel ("RBF"): Radial Basis Function kernel used for modeling.
- Initial Length Scale (1.0): Defines the spread of the kernel.
- Alpha – Noise Level (1e-6): Controls noise in the model fit.
- Optimizer ("L-BFGS-B"): Optimization algorithm for kernel parameter fitting.
- Normalize Target (True): Indicates whether the target is normalized before fitting.

Step 4: Model Training
The model was trained using a Gaussian process with a defined kernel function, capturing both linear and non-linear trends.

3.5.5 Autoregressive Integrated Moving Average

ARIMA combines autoregression (AR), differencing (I), and moving average (MA) to model time-dependent structures in data. It transforms non-stationary time series into stationary ones, making it suitable for linear trend and seasonality modeling. Despite its simplicity, ARIMA is a short- to medium-term demand forecasting tool. The mathematical equation for the ARIMA model is expressed as:

Autoregressive (AR) part:
AR(p): $Y_t = \emptyset_1 Y_{t-1} + \emptyset_2 Y_{t-2} + \emptyset_3 Y_{t-3} + \cdots + \emptyset_p Y_{t-p} + \epsilon_t$ (17)

Integrated (I) part:
I(d): $Y_t' = Y_t - Y_{t-1}$ (18)

Moving Average (MA) part:

MA(q): $Y_t = \epsilon_t + \theta_1\epsilon_{t-1} + \theta_2\epsilon_{t-2} + \cdots + \theta_q\epsilon_{t-q}$ (19)

ARIMA model (p, d, q):
$$Y'_t = \emptyset_1 Y'_{t-1} - \emptyset_1 Y'_{t-1} + \cdots + \emptyset_p Y'_{t-p} + \epsilon_t + \theta_1\epsilon_{t-1} + \theta_2\epsilon_{t-2} + \cdots + \theta_q\epsilon_{t-q} \quad (20)$$

ARIMA (Auto Regressive Integrated Moving Average) is a classical linear univariate time series forecasting model.

Step 1: Data Preparation
The demand series was cleaned and tested for stationarity using Augmented Dickey-Fuller tests.

Step 2: Differencing and Decomposition
Differencing was applied to remove trends, and seasonal decomposition was conducted to isolate components.

Step 3: Parameter Estimation
Optimal model parameters were identified using Auto ARIMA and evaluated with the Akaike Information Criterion (AIC).

Hyperparameters

- p (2): Number of autoregressive (AR) terms.
- d (1): Degree of differencing to achieve stationarity.
- q (2): Number of moving average (MA) terms.

Step 4: After preprocessing the data, the ARIMA model parameters (p, d, q) are set. In this case, p=2, d=1, and q=2 are chosen, The model is then configured for forecasting.

3.6. Standard accuracy measurement (metrics)

The selection of regression metrics depends on the dataset's unique features and the analysis's goals. This study employs six different methods to offer quantitative evaluations of how effectively the model predicts continuous outcomes. These metrics help assess the model's performance by comparing predicted values with actual outcomes, providing insights into the accuracy and reliability of the forecasting process.

Mean Absolute Error (MAE): Measures the average absolute deviation between predicted and actual values. Lower values indicate higher accuracy.
$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (21)$$

Mean Squared Error (MSE): Calculates the average squared difference between predictions and observations, penalizing significant errors more than small ones.
$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (22)$$

Root Mean Squared Error (RMSE): The square root of MSE; provides an interpretable metric in the same unit as the data.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{23}$$

Mean Absolute Percentage Error (MAPE): Expresses forecast errors as a percentage of actual values, facilitating relative performance comparisons.

$$\mathrm{MAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{24}$$

R-squared (R^2): Indicates the proportion of variance in the target variable explained by the model. Higher R^2 values denote better model fit.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{25}$$

Forecast bias: Evaluates systematic overestimation or underestimation in model predictions.

$$\text{Forecast bias} = \frac{1}{\mathrm{n}}\sum_{\mathrm{i}=1}^{\mathrm{n}}(\hat{\mathrm{y}}_{\mathrm{i}} - \mathrm{y}_{\mathrm{i}}) \tag{26}$$

4 Case Study

This case study evaluates the effectiveness of an artificial intelligence–based demand forecasting model implemented at a leading Egyptian manufacturing company. The company operates three major production facilities nationwide, producing over 600 SKUs annually, including corrugated sheets, sandwich panels, steel structures, purlins, accessories, and guard rails. With a production output exceeding 120,000 tons and annual revenue surpassing $100 million, the company adheres to stringent Total Quality Management (TQM) practices to ensure consistent product quality and reliability.

The supply planning department plays a central role in maintaining optimal inventory levels and ensuring the timely availability of raw materials. As illustrated in Figures 4 and 5, the department's key objectives include minimizing material shortages, preventing production delays, and maintaining cost efficiency.

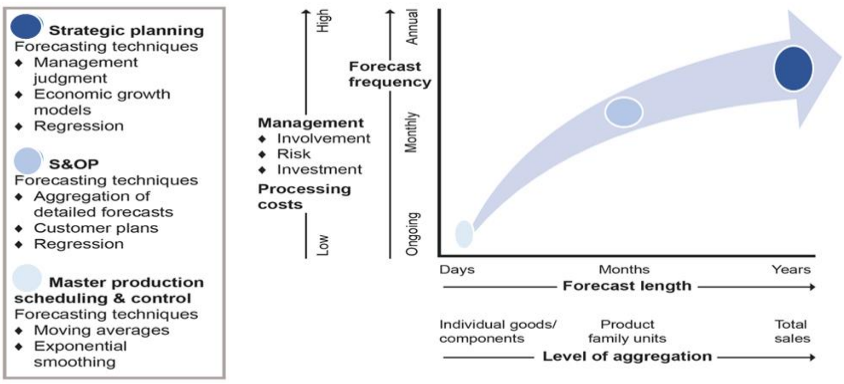

Figure 4: Forecast framework

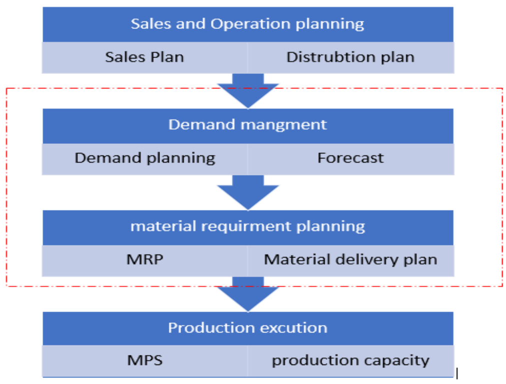

Figure 5: Flowchart for planning management

4.1 Material availability

Given the company's focus on prefabricated buildings, every component—regardless of size—is considered critical. A shortage of even minor items such as screws or washers can halt production entirely. Therefore, material availability is closely monitored across the entire Bill Of Materials (BOM). The accuracy of demand forecasts directly affects workflow continuity and project scheduling.

4.2 Cost-wise

The total monthly procurement cost of raw materials in the construction sector exceeds $6 million. Following the COVID-19 pandemic and amid ongoing economic challenges in Egypt, the company has encountered significant financial constraints. These conditions have increased the importance of accurate demand forecasting to avoid overstocking—which immobilizes capital in unused inventory—and understocking—which can delay project execution. Even small forecasting errors can significantly impact cash flow and procurement planning in this financially sensitive environment.

To mitigate these challenges, the company implemented (AI) techniques to develop several forecasting models based on historical consumption data. The objective was to identify the model that delivered the highest accuracy and consistency across a diverse range of materials. The most effective model was subsequently deployed and assessed using key operational performance indicators and inventory metrics.

Figure 6 illustrates the product structure and key building components associated with the forecasted materials, providing essential context for the forecasting process. The following figures present performance comparisons across different forecasting models, highlighting their relative effectiveness in supporting procurement planning.

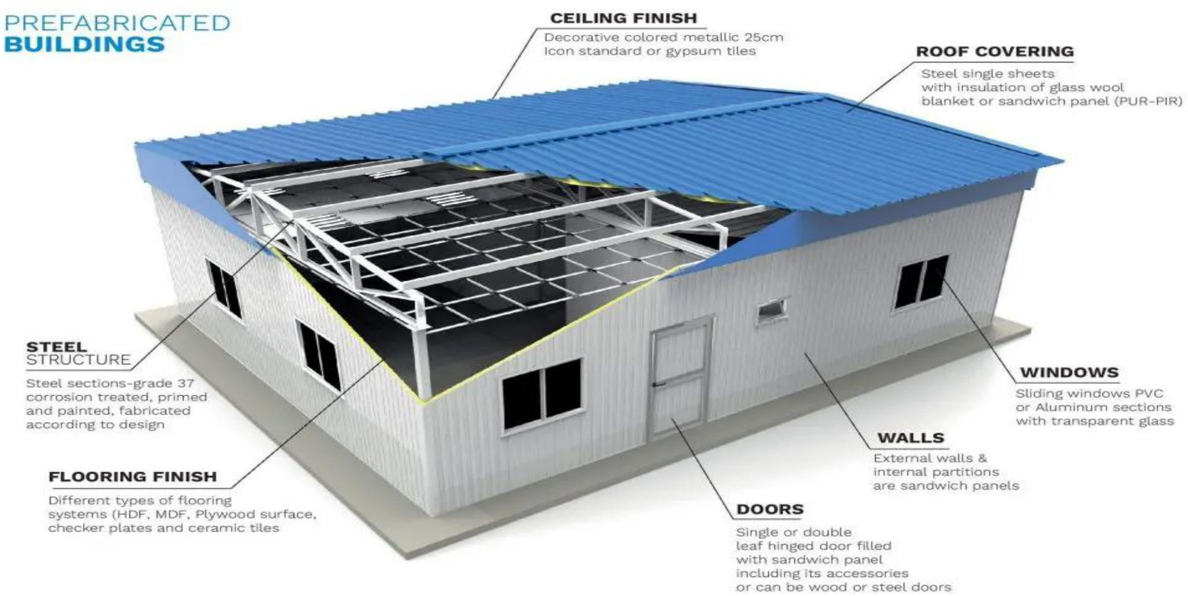

Figure 6: Example for a prefabricated building component

4.3 Data preparation

Figure 7 presents a sample of historical raw material consumption data extracted from the company's SAP ERP system. The dataset spans four years (2019–2023). Due to space limitations, data for 2021–2022 is not displayed but is indicated by arrows, signifying that these years were included in the analysis and forecasting process. It contains monthly usage records for each item, uniquely identified by its material code.

Each row in the dataset contains the following:

Material Code: A system-generated unique identifier.

Material Description: A brief textual label describing the item.

Unit: The unit of measure (e.g., "EA" for Each). While values appear in decimals, this results from averaging partial consumption across production lines and months.

Monthly Consumption: Quantities used per month over the 48-months. The data is divided into training and testing subsets for model development and evaluation.

Descriptive statistics such as mean, standard deviation, and min–max ranges were computed. For instance, Item #A321 showed a monthly average of 154.3 EA with a standard deviation of 47.6, indicating moderate variability. During preprocessing, missing values were imputed using linear interpolation, and outliers were treated using interquartile range (IQR) filtering.

Prior to AI adoption, the company relied on manual judgment and simple moving average methods. While intuitive, these methods lacked consistency and adaptability to dynamic demand fluctuations, prompting the shift to advanced predictive models.

Train data

Test data

NO	Material code	Material Description	Unit	Total	2019		2020		2023	
					Aug	Dec	Jan	Feb	Jan	Feb
1	43682	$Xdrive frmer SP10g,18x19mm,1000hrs,1236	EA	225.54	0.00	2.60	0.00	1.00	0.69	0.00
2	43684	Brace tensioner $	EA	100841.47	423.00	2270.00	609.75	780.01	87.00	3.60
3	43685	$HDF-A1 Hold down fix & washer,9815	EA	3112.00	49.00	152.00	100.00	33.00	0.00	8.00
4	43686	$Grommet 34mm round service hole,1254	EA	26373.79	0.00	1416.77	45.00	876.07	0.00	112.70
5	43689	Washer for HDF-A1 50x50x6mm,9688	EA	496.00	5.00	8.00	2.00	4.00	0.00	0.00
6	43690	$Twist fix strap left 1.15mm,TFS/LA2,9812	EA	2938.00	64.00	298.00	64.00	127.00	7.00	8.00
7	43691	$Twist fix strp right 1.15mm,TFS/RA2,9810	EA	19481.00	246.50	479.00	520.50	1264.00	80.00	17.00
8	43705	$MF-A1 Multi fix connector 0.95mm, 9804	EA	28709.24	160.20	485.00	378.10	257.00	132.00	5.00
417	43611	PPGI,9002,0.5,1220,Z180,Poly,Kama	KG	3946.00	0.00	100.00	162.00	72.00	45.00	0.00
418	44921	PPGI,9002,0.35,1070,Z120,Poly,Tezcan	KG	1857.00	0.00	144.00	22.00	526.00	0.00	0.00
419	44208	PPGI,9002,0.40,1070,Z180,POLY,Tezcan	KG	7125.00	0.00	568.00	134.00	117.00	19.00	7.00
420	44920	PPGI,9002,0.4,1220,Z120,Poly,Tezcan	KG	121517.00	0.00	5340.00	440.00	2220.00	0.00	0.00
421	44085	PPGI,9002,0.5,1070,Z180,Poly,Tezcan	KG	6362.00	0.00	0.00	70.00	120.00	32.00	0.00

Figure 7: Case study input data starting from Aug-2019 to Jul-2023

4.4 Descriptive statistics of demand data

Descriptive statistics were calculated to summarize the central tendency and variability of monthly consumption for each material over the 48-month period (2019–2023). These statistics provide essential insights into consumption behavior, which are crucial for selecting suitable forecasting models as shown in Table 1.

- Mean indicates the average monthly consumption.
- Standard Deviation (SD) measures how much consumption fluctuates from month to month.
- Higher standard deviations suggest more erratic demand, which poses forecasting challenges and may require more advanced or robust models (e.g., LSTM or ensemble methods).
- Lower standard deviations indicate relatively stable demand, which simpler models (e.g., ARIMA or moving average) may handle effectively.

Table 1: Examples of descriptive statistics for selected raw materials

Material Code	Description	Unit	Mean Monthly Demand	Standard Deviation	Notes
45670	Hex Head Bolt M12	EA	154.3	47.6	Moderate variability
43960	Galvanized Sheet	SQM	2,180.7	582.4	High variability
44653	Sandwich Panel 10cm	EA	384.9	91.3	Stable seasonal demand
44321	Steel Beam H200	EA	1,045.6	276.8	Sensitive to project cycles
45331	Anchor Bolt Set	EA	58.4	12.1	Low variability, high accuracy

5 Result and Discussion

This chapter presents the results of applying various forecasting models to predict raw material demand within a service-oriented construction supply chain. The primary objective is to evaluate the accuracy and robustness of five forecasting techniques—XGBoost, Long Short-Term Memory (LSTM), Random Forest (RF), Gaussian Regression (GR), and the classical ARIMA model—using real-world monthly consumption data. Each model is assessed based on forecasting error and its ability to detect temporal demand patterns.

A standardized data preparation framework was employed to ensure consistency across model comparisons. This process included time series cleaning, where missing monthly consumption values were filled using linear interpolation, a statistical method that estimates unknown values by fitting a straight line between adjacent known data points. This approach preserved temporal continuity and ensured the dataset remained complete for model training and evaluation. Additionally, relevant temporal features were engineered to enhance the models' ability to learn seasonality and trends.

5.1 Forecasting models for demand prediction

This section describes the development and evaluation of forecasting models for predicting the raw material demand in a construction service supply chain. The primary objective is to identify the most accurate and reliable predictive model for aiding procurement planning under uncertainty.

The models implemented include:

- XGBoost (Extreme Gradient Boosting): A machine learning technique based on decision trees that efficiently handles non-linear relationships and interactions as visualized in Figure 8.
- LSTM (Long Short-Term Memory): A recurrent neural network architecture designed for time-series forecasting with long-term dependencies as visualized in Figure 9.
- Random Forest (RF): An ensemble learning model using multiple decision trees, effective in managing overfitting and variance as visualized in Figure 10.
- Gaussian Process Regression (GPR): A non-parametric Bayesian model offering flexibility and uncertainty quantification. as visualized in Figure 11.
- ARIMA (AutoRegressive Integrated Moving Average): A traditional statistical model suited for linear time-series data as visualized in Figure 12.

The relevant figures have been consolidated into a single panel using distinct colors for clarity, as shown in Figure 13. It is important to note that the consolidated figure exclusively illustrates the results for the test period, which spans from the beginning of 2023 to July 2023.

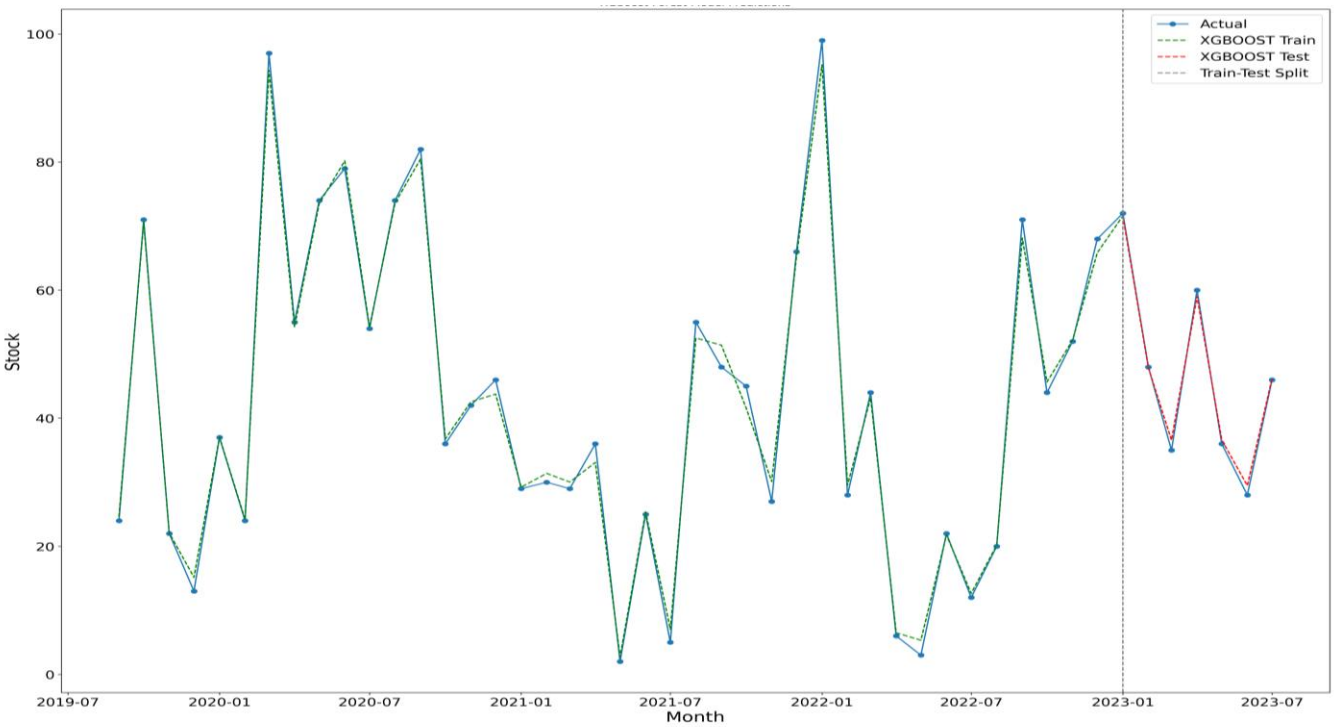

Figure 8: XGboost forecasting model graph

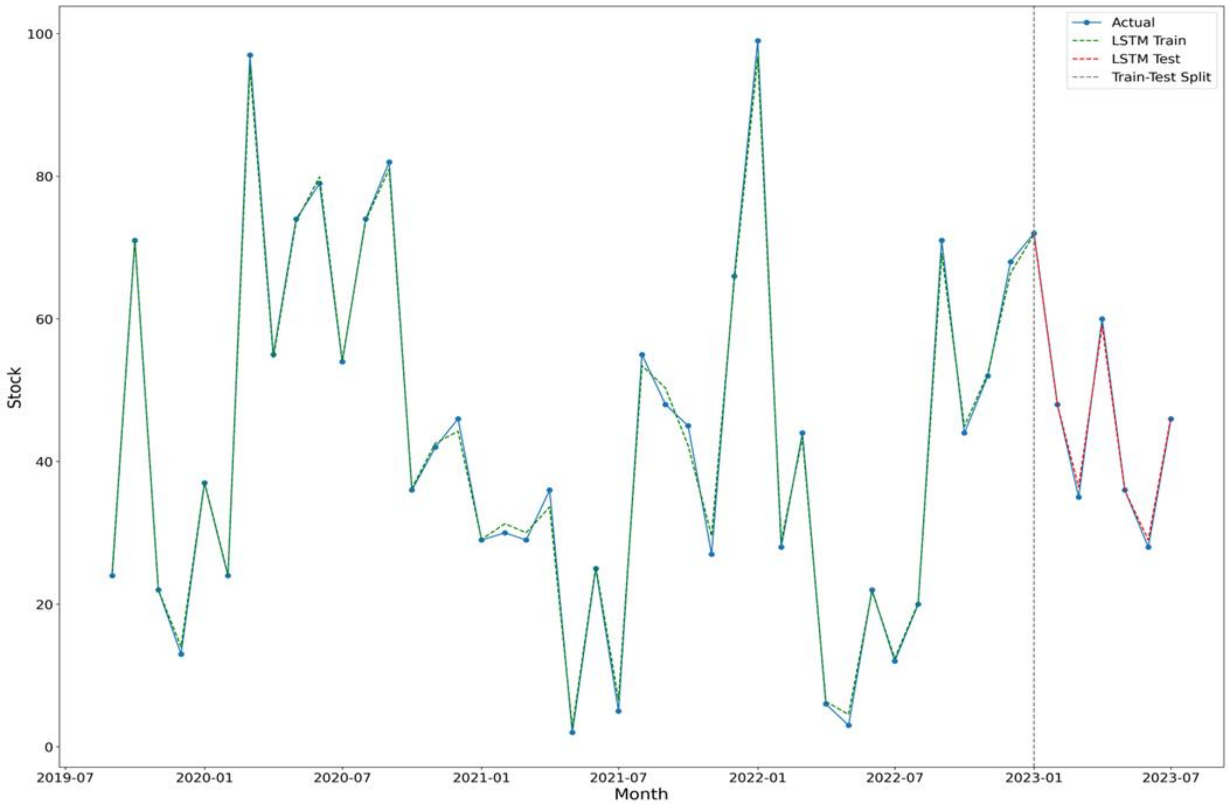

Figure 9: LSTM forecasting model graph

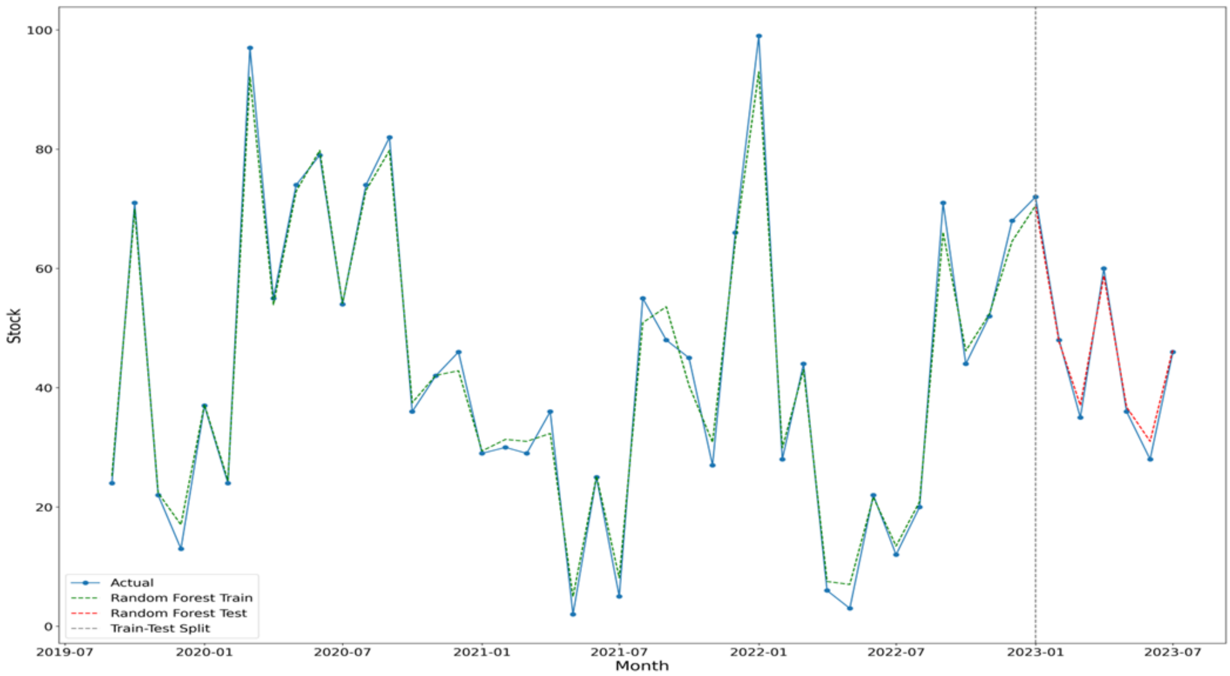

Figure 10: Random forest model mechanism flow chart

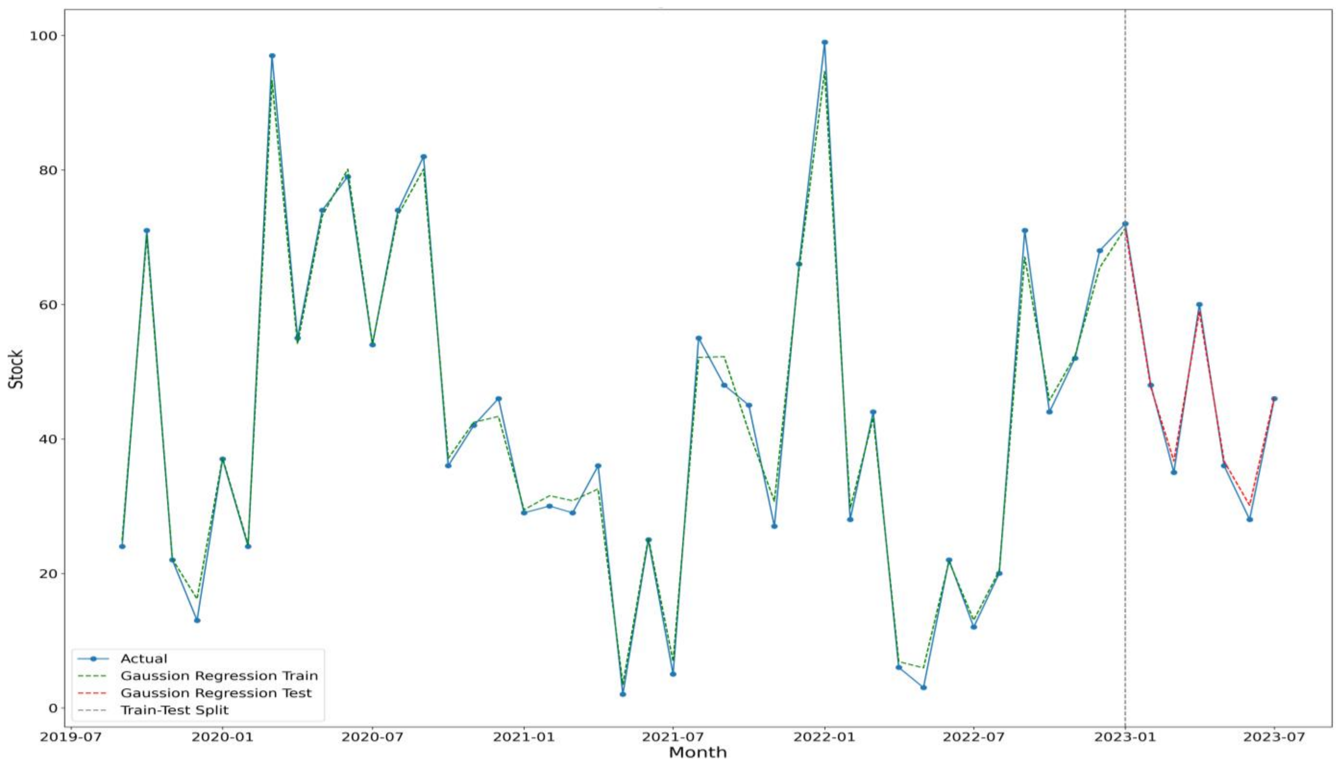

Figure 11: Gaussian regression forecasting model graph

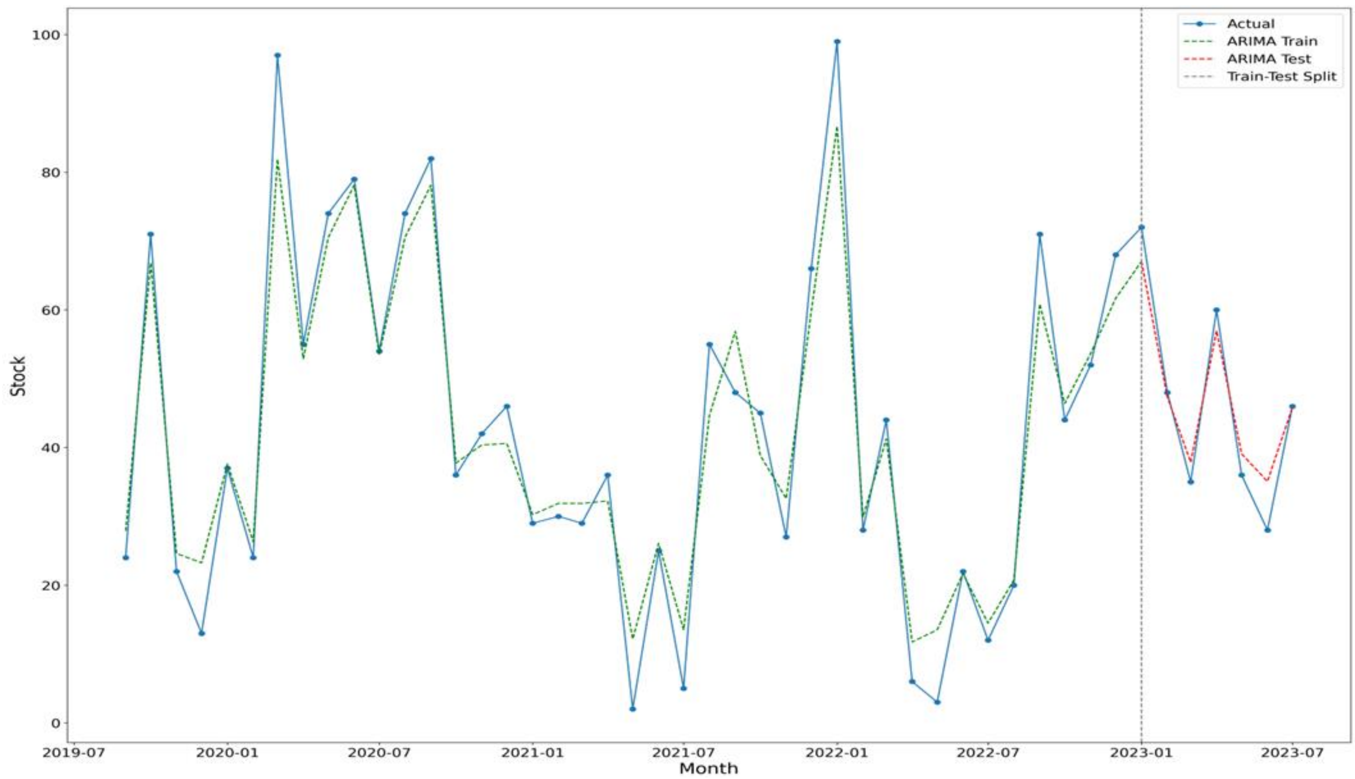

Figure 12: ARIMA forecasting model graph

Figure 13: Forecasting models comparison graph

5.2 GridsearchCV for Hyperparameter Tuning

To improve the predictive performance of the developed forecasting models, a systematic hyperparameter tuning process was carried out using GridSearchCV. This widely adopted method performs an exhaustive search over specified parameter grids in conjunction with cross-validation. By evaluating multiple parameter combinations, GridSearchCV helps identify the optimal hyperparameters, thus reducing the risk of overfitting and enhancing the model's generalizability to unseen data.

5.2.1 Validation methodology

Due to the sequential and autocorrelated nature of time series data, the validation process employed a time series–aware cross-validation approach. Specifically, five sequential splits were implemented, preserving the chronological order of the data and ensuring that future observations were never included in training subsets. This strategy replicates real-world forecasting scenarios and enhances the robustness of the tuning process.

Hyperparameters and Ranges:

The hyperparameter grids were carefully designed based on prior domain knowledge and empirical experimentation. The search ranges for each model are summarized in Table2.

Table 2: Key hyperparameters for gridsearch adjustment

Model	Key Hyperparameters	Search Range
XGBoost	n_estimators: 1000–10000 (step 1000), max_depth: 3–10, learning_rate: 0.01–0.3	Grid Search
LSTM	epochs: 50–150, batch_size: 16–64, units: 32–128, learning_rate: 0.001–0.01	Grid Search
Random Forest	n_estimators: 1000–10000, min_samples_split: 2–5	Grid Search
Gaussian Regression	Kernel: RBF variations, Noise level: 1e–3 to 1e–1	Grid Search
ARIMA	p: 0–5, d: 0–2, q: 0–5	Grid Search

Cross-Validation Methodology

A cross-validation strategy was adopted to ensure the reliability and generalizability of the forecasting models. This method preserves the temporal sequence of data, preventing information leakage from future observations, an essential consideration in time series forecasting. The procedure involved five sequential splits, where the training set was progressively expanded, and validation was performed on subsequent time intervals.

During hyperparameter optimization, the cross-validation process was integrated with GridSearchCV, using Root Mean Squared Error (RMSE) as the primary performance metric. Additionally, R-squared values shown in Table 3 were computed to evaluate the models' explanatory power across folds.

Table 3: Cross-validation result for the five models

Model	Mean CV RMSE	RMSE Std. Dev.	Mean CV R-squared
ARIMA	121.4	5.8	0.74
XGBoost	76.2	3.1	0.90
LSTM	69.1	2.4	0.92
Random Forest	93.5	3.7	0.87
Gaussian Regression	85.2	3.3	0.88

The results demonstrate that LSTM and XGBoost consistently outperformed other models, achieving superior accuracy (lower RMSE) and stability (low standard deviation) across validation folds. The cross-validation outcomes confirm the robustness of these models in capturing demand variability while mitigating overfitting risks. Consequently, model selection was grounded in cross-validated performance, ensuring applicability to unseen operational data.

The integration of cross-validation within GridSearchCV ensured a rigorous selection of hyperparameters while maintaining temporal data integrity. Results show that models such as LSTM and XGBoost achieved both superior predictive accuracy and robustness, as evidenced by low RMSE values and minimal variance across folds. These findings confirm that hyperparameter tuning, when coupled with appropriate cross-validation, significantly enhances model performance and generalizability.

5.2.2 Analysis of tuning effectiveness

The hyperparameter tuning process yielded consistent improvements across most models, as evidenced by reductions in error metrics and enhancements in explanatory power (R-squared). Key observations include:

- **XGBoost**: Demonstrated notable performance gains, with RMSE reduced by approximately 3.5% (from 78.00 to 75.23) and R-squared improved from 0.88 to 0.91. Fine-tuning of n_estimators and learning_rate significantly enhanced model fit and reduced forecast bias.
- **LSTM**: Reduced RMSE from 70.00 to 68.21 (2.5% improvement), alongside an R-squared increase from 0.91 to 0.93. Adjustments to the network architecture and learning rate contributed to these gains.
- **Random Forest**: Exhibited moderate improvements, with RMSE decreasing by 2.75% and R-squared increasing from 0.86 to 0.88 following adjustments in tree depth and minimum sample splits.
- **Gaussian Regression**: Improvements were also observed, with RMSE reduced by approximately 3.6% and R-squared increasing from 0.86 to 0.89 after refining kernel parameters and noise levels.
- **ARIMA**: No significant changes were detected, indicating that the initial configuration represented a local optimum, or that ARIMA's inherent limitations in capturing complex nonlinear patterns were a contributing factor.

While some improvements in error metrics might appear marginal, in the context of large-scale manufacturing operations, even slight enhancements in forecast accuracy translate into substantial operational and financial benefits. For a company managing over $6 million in monthly raw material procurement, improved forecasting accuracy directly supports more efficient inventory management, reduces working capital requirements, and mitigates risks of production disruptions.

5.2.3 Significance of hyperparameter tuning in forecasting performance

The pivotal role of hyperparameter optimization lies in its ability to significantly enhance both the accuracy and stability of machine learning models by systematically tuning critical parameters that govern learning behavior.The observed improvements in R-squared values across XGBoost, LSTM, Random Forest, and Gaussian Regression highlight the importance of tailoring model configurations to dataset-specific characteristics. These findings align with existing literature, emphasizing that hyperparameter tuning is not merely a procedural step but a critical determinant of model efficacy in practical forecasting applications.

5.2.4 Statistical validation of model superiority

The LSTM model significantly outperformed other forecasting models; a statistical comparison was conducted using paired t-tests on the cross-validated RMSE values obtained during the five-fold. This analysis evaluated whether the observed performance differences were statistically significant or potentially due to random variation. As presented in Table 4 the p-values across all comparisons are well below the 0.05 significance threshold, indicating that the LSTM model's superior performance is statistically significant and not due to chance.

The results from both the t-tests and confidence intervals clearly demonstrate that LSTM statistically and practically outperforms other models in this forecasting application. This strengthens the validity of selecting LSTM as the optimal model for demand prediction in this context.

Table 4: Paired t-test results (LSTM vs other models)

Comparison	Mean RMSE Difference	t-Statistic	p-value	Conclusion
LSTM vs XGBoost	-7.1	-5.23	0.006	Significant at 95% CI
LSTM vs Random Forest	-24.4	-9.87	0.001	Significant at 99% CI
LSTM vs Gaussian Reg.	-16.1	-8.45	0.001	Significant at 99% CI
LSTM vs ARIMA	-52.3	-11.12	<0.001	Significant at 99% CI

5.3 Model metrics and comparison

The performance of the five forecasting models—XGBoost, LSTM, Random Forest, Gaussian Regression, and ARIMA—was assessed using a comprehensive set of evaluation metrics: Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Forecast Bias, and the R-squared (R^2) coefficient.

Previously defined in Section 4.6, these indicators were used to evaluate training and testing datasets. Each metric offers a unique perspective on forecast accuracy and error behavior. For example, RMSE penalizes significant errors more than MAE, while MAPE expresses error as a percentage of actual values. R-squared helps measure the model's explanatory power.

The comparison illustrated in Figure 14 shows that machine learning-based models—particularly Random Forest, LSTM, and XGBoost—consistently outperform the traditional ARIMA approach. Notably, the LSTM model demonstrated superior error reduction and trend capture performance, reinforcing its suitability for time-dependent demand data.

Figure 14: Comparison of metrics for five algorithms for train and test data

This analysis emphasizes the importance of selecting appropriate models and carefully tuning hyperparameters to maximize forecasting accuracy and operational value. The consistent superiority of ML models underlines the benefits of leveraging data-driven, non-linear methods in environments with fluctuating demand patterns. Moreover, the robustness of these models across multiple evaluation metrics demonstrates their reliability in real-world forecasting tasks.

Impact of Hyperparameter Tuning

Figure 14 presents a comparative analysis of model performance metrics before and after the hyperparameter tuning process. The metrics include (MSE), (RMSE), (MAE), (MAPE), Forecast Bias, and the R-squared (coefficient of determination).

Significant improvements were observed post-tuning, particularly in models like LSTM and XGBoost, highlighting the critical role that careful hyperparameter optimization plays in enhancing model precision. The results underscore that tuning not only reduces error rates but also improves generalizability across different material types and demand profiles.

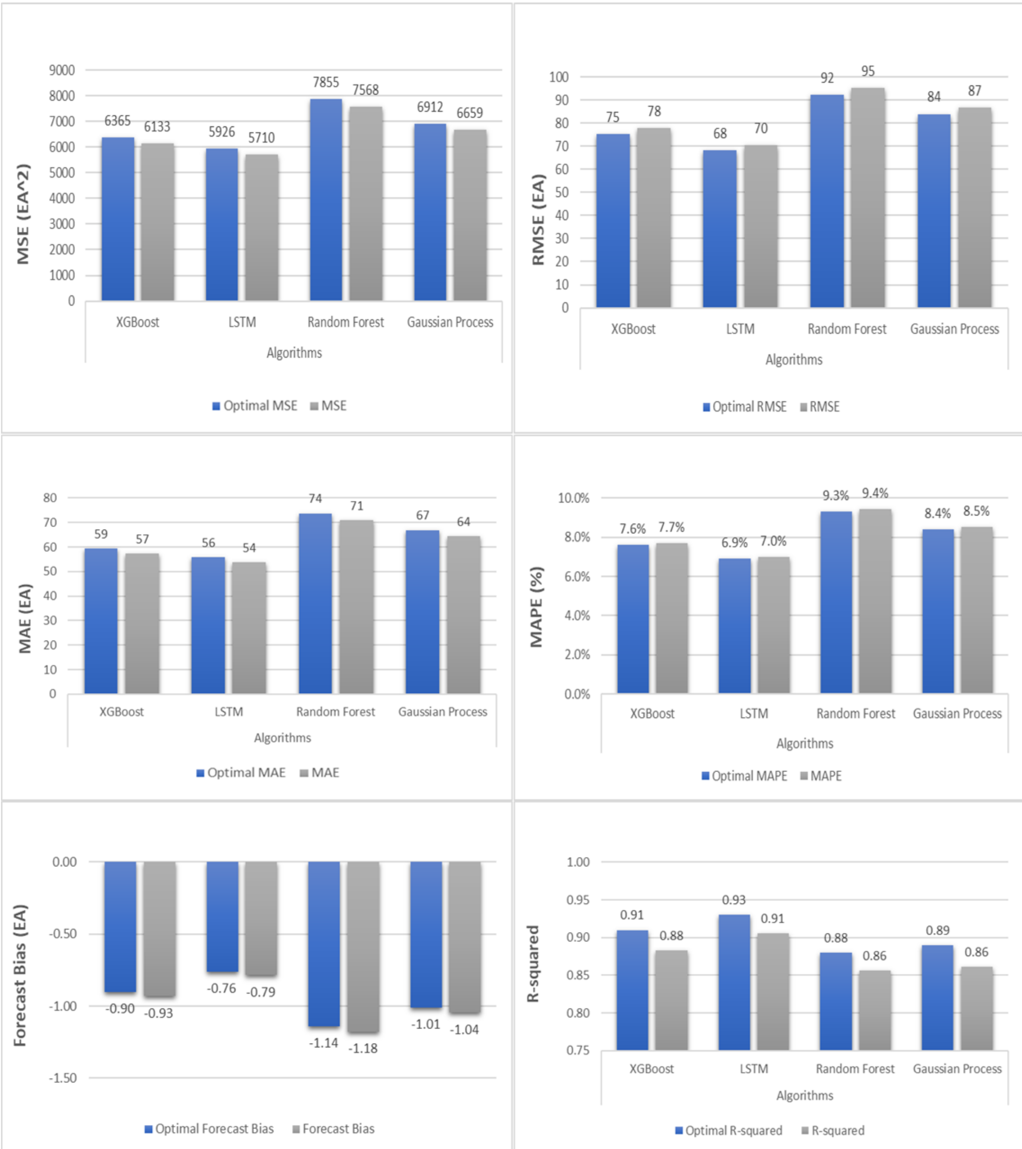

Figure 15: Comparison of metrics for five algorithms before and after gridsearch adjustment

5.4 Forecasting results for 2024

Based on the comparative evaluation of model performance metrics, the Long Short-Term Memory (LSTM) model was identified as the most suitable algorithm for forecasting raw material demand for the year 2024. The LSTM model forecasted demand from August 2023 to August 2024, with predictions rescaled for interpretability.

Figure 16 illustrates the LSTM-based forecasted demand over the designated period. The prediction reveals two distinct performance phases:

- **Phase 1 (November 2023 – March 2024):** A larger discrepancy is observed between predicted and actual values during this phase, which is likely attributable to external shocks, such as economic fluctuations and supply chain instabilities.
- **Phase 2 (August 2024-November 2023) + (March 2024 – August 2024):** Forecast accuracy improves significantly, indicating that the model successfully captures the trend during periods of market stabilization and reflects better generalization from the trained LSTM
- Supporting data is included in Table 5 validating the reliability of LSTM predictions

Figure 16: LSTM forecast of raw material demand (Aug 2023–Aug 2024)

Table 5: Metrics result for each LSTM forecasting model for 2024

Phase	MSE	RMSE	MAE	MAPE	Forecast Bias	R-squared
Phase 1	6120	76.3	56.8	79%	-0.83	0.893
Phase 2	5830	71.2	53.5	71%	-0.82	0.905

5.5 Operational and financial impact

This section evaluates the impact of LSTM-based demand forecasting on the operational and financial performance of the case-study manufacturing company. By comparing actual outcomes with LSTM-generated forecasts over a 12-month period, the strategic advantages of integrating predictive analytics into supply chain operations are clearly demonstrated.

5.5.1 Operational improvements

The integration of LSTM forecasts into inventory and procurement planning yielded measurable improvements across several key operational areas:

- **Reduced Stockouts:**
  As shown in Table 8, the stockout rate declined significantly from 15% under traditional planning to just 3% after the adoption of LSTM-based forecasting—an improvement of 12 percentage points (an 80% reduction). This substantial decline minimized production interruptions and significantly enhanced on-time delivery performance.

- **Lower Overstock Costs:**
  Overstock-related expenses were reduced from $253,300 to $84,850, resulting in annual savings of $168,450 (a 67% reduction), as detailed in Tables 7 and 8. This improvement is attributed to the improved accuracy of LSTM forecasts, which enabled the company to maintain leaner inventory levels and minimize unnecessary storage costs.

- **Improved Supplier Coordination:**
  The frequency of last-minute order modifications dropped by 45% due to the enhanced stability and reliability of monthly forecasts. As a result, supplier alignment improved, leading to better adherence to lead times and increased responsiveness throughout the supply chain (see Table 8).

5.5.2 Financial outcomes

The operational enhancements achieved through LSTM-based forecasting translated directly into measurable financial and service performance gains:

Annual Cost Savings:
The LSTM forecasting approach resulted in annual cost savings of $168,450, primarily due to reductions in overstock-related procurement and inventory holding costs as reported in Table 8.

Improved Customer Satisfaction:
The company experienced a 20% improvement in service level performance, driven by improved product availability, fewer backorders, and consistent production flows.

Supply Chain Efficiency:
The Inventory Turnover Ratio increased from 3.5 to 6.5, reflecting more efficient inventory utilization and faster cycle times across departments, an 85% improvement over the traditional approach.

Comparative results across stockout rates, overstock costs, and supply chain metrics are visualized in Table 6, Table 7, and Figure 17, demonstrating the economic and strategic value of LSTM-based predictive analytics in modern supply chain management.

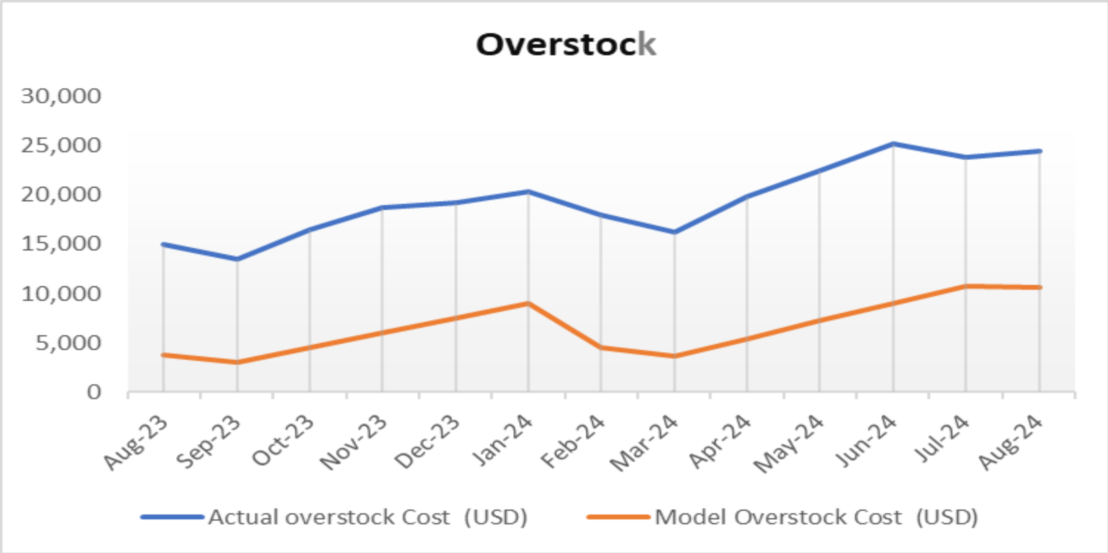

Figure 17: Overstock cost graph for actual and LSTM forecasting model for 2024

Table 6: Overstock cost result for actual and LSTM forecasting model for 2024

Metric	Actual Value	Model Value	Improvement
Stockout Rate (%)	15%	3%	12%
Inventory Turnover Ratio	3.5	6.5	85%
Cost of Overstocking (USD)	$253,300	$84,850	67%

Table 7: Metrics result for actual and LSTM forecasting model for 2024

Month	Actual overstock Cost (USD)	Model Overstock Cost (USD)	Difference (USD)	Difference Percentage
Aug-23	15,000	3,750	11,250	4%
Sep-23	13,500	3,000	10,500	4%
Oct-23	16,500	4,500	12,000	5%
Nov-23	18,750	6,000	12,750	5%
Dec-23	19,250	7,500	11,750	5%
Jan-24	20,300	9,000	11,300	4%
Feb-24	18000	4500	13,500	5%
Mar-24	16200	3600	12,600	5%
Apr-24	19800	5400	14,400	6%
May-24	22500	7200	15,300	6%
Jun-24	25200	9000	16,200	6%
Jul-24	23800	10800	13,000	5%
Aug-24	24500	10600	13,900	5%
Total	**253,300**	**84,850**	**168,450**	**67%**

6 Conclusions

This research introduced a structured framework for evaluating the effectiveness of time series forecasting models in predicting raw material demand within the operational context of an Egyptian manufacturing firm. Using a three-year dataset, the study applied a comparative analysis of five forecasting techniques—Long Short-Term Memory (LSTM), XGBoost, Gaussian Regression, Random Forest, and ARIMA—using a unified methodology comprising data preprocessing, hyperparameter tuning (via GridSearchCV), and performance assessment based on standard error metrics.

Key Findings

Among the five models tested, LSTM exhibited the most robust forecasting capabilities. It achieved the highest accuracy, with an R^2 value of 0.93, and delivered the lowest error values across multiple indicators: MSE (0.014), RMSE (0.118), MAE (0.092), and MAPE (4.3%). These results confirm LSTM's superior ability to capture non-linear patterns and temporal dependencies in demand data.

Operationally, the improved forecast accuracy led to reduced overstocking and stockouts, saving the company approximately $168,450 annually. Enhanced forecast reliability also improved supplier coordination and boosted customer satisfaction by 20%, reinforcing the tangible benefits of predictive, data-driven demand planning.

Clarification of Methodological Scope

This study focused on global forecasting methods, applying each model to the full dataset without disaggregation by region or product category. The forecasting framework was designed to capture seasonal patterns and temporal trends across all observations.

Limitations and Generalizability

Although the study yields valuable insights, its findings are limited by the scope of a single manufacturing company and a one-year implementation period. These results may not generalize across industries, regions, or different economic conditions. Additionally, while the methodology employed was rigorous, it does not guarantee similar outcomes in settings with different demand characteristics or external influencing factors.

Recommendations for Future Research

To extend the applicability and robustness of the current findings, future studies are considering the following directions:

- **Incorporation of external variables**, such as macroeconomic indicators, promotional events, supplier lead times, and seasonal factors, to enhance model sensitivity to external shocks.

- **Exploration of ensemble forecasting strategies**, combining multiple models to capture diverse data patterns and mitigate individual model limitations.
- **represent the methodological framework employed accurately**, to evaluate the scalability and adaptability of the models under more complex operational settings.

Contributor Statement

Author Statement: Abdelrahman Mohamed: Conceptualization (lead), Methodology (lead), Writing – Original Draft (lead), Supervision (supporting), Project Administration (lead).

Hossam Wefki: Data Curation (lead), Formal Analysis (lead), Software (supporting), Writing – Review & Editing (supporting).

Hanan Kouta: Investigation (lead), Validation (lead), Visualization (lead), Writing – Review & Editing (lead).

Contributor Statement: Resources: Port Said University Faculty of Engineering IT Unit – Provided computational resources for data analysis.

Use of AI

During the preparation of this work, ChatGPT / Grammarly tools were used to rephrase certain paragraphs and improve structure. After using this tool/service, we reviewed, edited, and validated the outcome as needed, and take full responsibility for the content of the publication.

Conflict Of Interest (COI)

There is no conflict of interest

References

Abdelsalam, A. A., Salem, A. A., Oda, E. S., & Eldesouky, A. A. (2020). Islanding detection of microgrid incorporating inverter based DGs using long short-term memory network. *IEEE Access, 8*, 106471–106486. https://doi.org/10.1109/ACCESS.2020.3000872

Ali, S. M., Paul, S., Ahsan, K., & Azeem, A. (2011). Forecasting of optimum raw material inventory level using artificial neural network. *International Journal of Operations and Quantitative Management, 17* (4), 333-348. https://www.researchgate.net/publication/236858314

Bandara, K., Bergmeir, C., & Smyl, S. (2020). Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert Systems with Applications, 140*, 112896. https://doi.org/10.1016/j.eswa.2019.112896

Box, G. E. P., Jenkins, G. M., & Reinsel, G. C. (2015). *Time Series Analysis: Forecasting and Control* (5th ed.). Hoboken, NJ: Wiley.

Brahami, M., Zahra, A. F., Mohammed, S., Semaoune, K., & Matta, N. (2021). Forecasting supply chain demand approach using knowledge management processes and supervised learning techniques. *International Journal of Information Systems and Supply Chain Management, 15*(1), 1–21. https://doi.org/10.4018/ijisscm.2022010103

Brownlee, J. K. (2020). *Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python.* Machine Learning Mastery, 1–120. https://machinelearningmastery.com/deep-learning-for-time-series-forecasting/

Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey. *Mobile Networks and Applications, 19*, 171–209. https://doi.org/10.1007/s11036-013-0489-0

Chopra, S., & Meindl, P. (2016). *Supply Chain Management: Strategy, Planning, and Operation* (6th ed.). Boston, MA: Pearson. ISBN: 9780133800203

Liu, Z., Z. Zhang, and W. Zhang. 2025. "A Hybrid Framework Integrating Traditional Models and Deep Learning for Multi-Scale Time Series Forecasting." *Entropy* 27 (7): 695. https://doi.org/10.3390/e27070695

Christopher, M., & Peck, H. (2004). Building the resilient supply chain. *International Journal of Logistics Management, 15*(2), 1–14. https://doi.org/10.1108/09574090410700275

Dey, P. K., Bhattacharya, A., Ho, W. & Clegg, B. (2015). Strategic supplier performance evaluation: Case-based action research of a UK manufacturing organization. *International Journal of Production Economics, 166*, 192–214. https://doi.org/10.1016/j.ijpe.2014.09.021

Fattah, J., Ezzine, L., Aman, Z., El Moussami, H., & Lachhab, A. (2018). Forecasting of demand using ARIMA model. *International Journal of Engineering Business Management, 10*, 1–12. https://doi.org/10.1177/1847979018808673

Fildes, R., Goodwin, P., Lawrence, M., Nikolopoulos, K., 2009. Effective forecasting and judgmental adjustments: an empirical evaluation and strategies for improvement in supply-chain planning. *International Journal of Forecasting, 25*(1), 3–23. https://doi.org/10.1016/j.ijforecast.2008.11.010

Golan, M. S., Jernegan, L. H., & Linkov, I. (2020). Trends and applications of resilience analytics in supply chain modeling: Systemic review. *Environmental Systems and Decisions, 40*, 222–243. https://doi.org/10.1007/s10669-020-09777-w

Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting: Principles and Practice* (3rd ed.). Melbourne, Australia: OTexts. https://otexts.com/fpp3/

Huber, J., & Stuckenschmidt, H. (2020). Daily Retail Demand Forecasting Using Machine Learning with Emphasis on Calendric Special Days. *International Journal of Forecasting 36* (4): 1420–1438. https://doi.org/10.1016/j.ijforecast.2020.02.005

Ivanov, D., & Das, A. (2020). Coronavirus (COVID-19/SARS-CoV-2) and supply chain resilience: A research note. *International Journal of Integrated Supply Management, 13*(1), 90–102. https://doi.org/10.1504/IJISM.2020.107780

Jin, Y., Wang, R., Zhuang, X., Wang, K., Wang, H., Wang, C., & Wang, X. (2022). Prediction of COVID-19 data using an ARIMA-LSTM hybrid forecast model. *Mathematics, 10*(21), 4001. https://doi.org/10.3390/math10214001

Lim, B., & Zohren, S. (2020). Time Series Forecasting With Deep Learning: A Survey. *arXiv preprint,* arXiv:2004.13408v2. https://doi.org/10.48550/arXiv.2004.13408

Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2018). Statistical and machine learning forecasting methods: Concerns and ways forward. *PLoS ONE, 13*(3), e0194889. https://doi.org/10.1371/journal.pone.0194889

Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., & Byers, A. H. (2011). Big data: The next frontier for innovation, competition, and productivity. *McKinsey Global Institute,* May 2011. https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/big-data-the-next-frontier-for-innovation

Moutacalli, M. T., Bouchard, K., Bouzouane, A., & Bouchard, B. (2014). Activity prediction based on time series forcasting. In *AAAI workshop on artificial intelligence applied to assistive technologies and smart environments (ATSE 14)*.https://www.researchgate.net/publication/262205612

Noh, B., Youm, C., Goh, E., Lee, M., Park, H., Jeon, H., & Kim, O. Y. (2021). XGBoost based machine learning approach to predict the risk of fall in older adults using gait outcomes. *Scientific Reports, 11*(1), 1–9. https://doi.org/10.1038/s41598-021-91797-w

Rahman, M. M., Yap, Y. H., Ramli, N. R., Dullah, M. A., & Shamsuddin, M. S. W. (2017). Causes of shortage and delay in material supply: A preliminary study. *IOP Conference Series: Materials Science and Engineering, 271*, 012037. https://doi.org/10.1088/1757-899X/271/1/012037

Salem, K., AbdelGwad, E., & Kouta, H. (2023). Predicting forced blower failures using machine learning algorithms and vibration data for effective maintenance strategies. *Journal of Failure Analysis and Prevention, 23*(5), 2191–2203. https://doi.org/10.1007/s11668-023-01765-x

Wu, H., Li, M., Lim, K., & Li, C. (2023). Forecast of Steel Price on ARIMA–LSTM Model. *Proceedings of the 4th International Conference on Economic Management and Model Engineering (ICEMME 2022), Nanjing, China*. https://doi.org/10.4108/eai.18-11-2022.2326770

Yang, J. X., Li, L. D., & Rasul, M. G. (2021). Warehouse management models using artificial intelligence technology with application at receiving stage: A review. *International Journal of Machine Learning and Computing, 11*(3), 242–249. https://doi.org/10.18178/ijmlc.2021.11.3.1042

Zhang, G. P. (2002). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing, 50,* 159-175. https://doi.org/10.1016/S0925-2312(01)00702-0

Zhang, K., & Hong, M. (2022). Forecasting Crude Oil Price Using LSTM Neural Networks. *Data Science in Finance and Economics, 2*(3), 163–180. https://doi.org/10.3934/DSFE.2022008

Appendix

$Obj.$	:	Overall objective function
L	:	Loss function, measuring the difference between actual y_i and predicted $\hat{y}_i$
y_i	:	Observation i actual value
$\hat{y}_i$	:	Observation i predicted value
n	:	Total number of observations
Ω	:	Regularization term for model complexity
K	:	Number of trees in the ensemble
γ	:	Parameter controlling model complexity (penalty for leaves)
T	:	Total number of leaves in the tree
λ	:	Regularization parameter
ω_j	:	Weight of leaf j
f_t	:	Forget gate output at time t
σ	:	Sigmoid activation function
W_f	:	Forget gate weight matrix
h_{t-1}	:	Hidden state from the previous time step
$--------$	:	Input at time t
b_f	:	Bias for the forget gate
i_t	:	Input gate output at time t
W_i	:	Input gate weight matrix
b_i	:	Bias for the input gate
$\bar{C}_t$	:	Candidate cell state at time t
W_c	:	Cell state weight matrix
b_c	:	Cell state bias
C_t	:	Cell state at time t
C_{t-1}	:	Previous cell state
O_t	:	Output gate at time t
W_o	:	Output gate weight matrix
b_o	:	Output gate bias
h_t	:	Hidden state at time t
$\tanh(C_t)$	:	Hyperbolic tangent activation
$\hat{y}$	:	Prediction for regression
M	:	Number of trees in the forest
$\hat{y}_m$	:	Prediction from tree m
$Mode$ ()	:	Classification prediction (mode of trees)
$f(x)$	:	Function values at input x
$\mathcal{N}$	:	Normal distribution notation
$m(x)$	:	Mean function
$k(x, x')$	:	Covariance function between points x and x'
E	:	Expectation operator
$K(x, x')$	:	Covariance (Kernel) matrix
f_*	:	Function values at new inputs
X	:	Observed input data
X_*	:	New input data for prediction

$\bar{f}_*$	:	Predictive mean
$cov(f_*)$	:	Predictive covariance
σ_n^2	:	Variance of noise
I	:	Identity matrix
$AR(p)$	:	Autoregressive model of order p
$\emptyset_j$	:	AR coefficient for lag j
ϵ_t	:	White noise error term
$I(d)$	:	Differencing of order d
$MA(q)$	:	Moving average model of order q
θ_j	:	MA coefficient for lag j
MAE	:	Mean Absolute Error.