Archives available at journals.mriindia.com

Multidisciplinary Journal of Research in Engineering and Technology

ISSN: 2348-6953

Volume 13 Issue 01, 2026

Intelligent Predictive Optimization Framework for Demand Forecasting and Inventory Planning in Food Supply Chains

[1]Mohd Abdul Razzaque Mohd Abdul Rasheed, [2]Syed Abrar Azhar, [3]V. S. Karwande
[1]Computer Science and Engineering Department, Everest College of engineering and technology, chhatrapati Sambhajinagar
[2,3] Computer Science and Engineering Department, Everest College of engineering and technology, chhatrapati Sambhajinagar[2]
[1]abdul.razzaquecse@gmail.com, [2]hodcse@eescoet.org, [3]vijayskarwande@gmail.com

Peer Review Information

Submission: 15 April 2026
Revision: 06 May 2026
Acceptance: 25 May 2026

Keywords

Food Supply Chain Management, Demand Forecasting, Predictive Optimization, Time Series Forecasting, Machine Learning, XGBoost, Long Short-Term Memory, Temporal Fusion Transformer, Inventory Planning, EOQ, Newsvendor Model, Linear Programming, Decision Support Dashboard, MAPE, RMSE

Abstract

Food supply chain management is highly challenging because demand patterns change due to seasonality, weather conditions, promotions, consumer behaviour, and product perishability. Traditional forecasting methods often fail to capture these nonlinear and time-dependent variations, which may lead to inaccurate demand estimation, excess stock, stockouts, and food wastage. To address these issues, this paper presents an intelligent predictive optimization framework for demand forecasting and inventory planning in food supply chains. The proposed framework processes multi-source data, including historical sales records, weather variables, promotional information, and product-level demand trends. It applies time series and machine learning models such as ARIMA, XGBoost, Long Short-Term Memory, and Temporal Fusion Transformer to generate multi-horizon demand forecasts. These forecast outputs are then used by optimization models, including Economic Order Quantity, Newsvendor formulation, and Linear Programming, to support cost-efficient replenishment and inventory planning while maintaining required service levels. The framework also includes a decision-support dashboard that allows planners to visualize forecast trends, compare demand patterns, monitor inventory indicators, and perform scenario-based analysis. The system performance is evaluated using standard forecasting metrics such as Mean Absolute Percentage Error, Root Mean Square Error, and Bias. The proposed approach connects predictive analytics with prescriptive optimization, enabling more accurate demand planning, better stock control, reduced wastage, and improved operational efficiency. Overall, the framework provides a scalable and data-driven solution for sustainable and responsive food supply chain management.

Introduction

The modern food supply chain works as a highly connected and data-dependent system in which producers, suppliers, distributors, warehouses, and retailers must coordinate continuously to meet changing consumer demand. Demand in the food sector is difficult to predict because it is affected by several factors, including seasonality, promotional offers, price changes, festivals, weather conditions, and consumer buying behaviour. Because food products are often perishable, maintaining the right balance

© 2026 The Authors. Published by MRI INDIA.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

between product availability and minimum wastage becomes an important operational challenge.

Traditional forecasting methods, which are commonly based on simple statistical techniques or manual planning, are often not sufficient for such dynamic demand conditions. These methods may not capture nonlinear demand behaviour, external influencing factors, and long-term temporal dependencies effectively. As a result, demand estimates become less reliable. Poor forecasting can lead to stockouts, lost sales, and reduced customer satisfaction. On the other hand, excess inventory increases storage cost and contributes to food wastage.

To overcome these limitations, Artificial Intelligence and Machine Learning techniques have become important in supply chain management. These methods can learn from historical sales data, identify hidden demand patterns, and adapt to changing market conditions. When forecasting models are combined with optimization techniques, the system can move beyond demand prediction and support better inventory decisions. This connection between forecasting and optimization forms the base of an intelligent decision-support system for food supply chain planning. [1]

However, many existing systems still treat demand forecasting and inventory optimization as separate tasks. This separation reduces the practical value of forecasting because predicted demand is not always converted into suitable replenishment decisions. In addition, limited use of multi-source data, weak integration of external factors, and lack of decision interpretability make such systems difficult to adopt in real food supply chain environments. [4]

To address these issues, this work presents a unified framework that integrates multivariate demand forecasting with optimization-based inventory planning. The system uses multiple forecasting models, including ARIMA, XGBoost, LSTM, and Temporal Fusion Transformer, to capture different demand patterns across time. ARIMA supports statistical time-series modelling, XGBoost captures structured feature-based relationships, LSTM learns sequential demand behaviour, and Temporal Fusion Transformer supports multi-horizon forecasting using temporal and contextual inputs. The generated forecasts are then passed to optimization models such as EOQ, Newsvendor, and Linear Programming to support cost-effective replenishment and service-level-oriented inventory decisions.

The main contribution of this work is the development of an end-to-end predictive optimization pipeline that connects data ingestion, preprocessing, forecasting, inventory optimization, and dashboard visualization. This integrated structure ensures that inventory decisions are based on forecasted demand, contextual factors, and operational constraints. The decision-support interface further helps planners visualize demand trends, compare forecast outputs, monitor inventory indicators, and evaluate scenario-based decisions. This improves planning accuracy, operational efficiency, and sustainability in the food supply chain. [5]

In summary, the proposed framework bridges the gap between predictive analytics and prescriptive decision-making. By linking demand forecasting with inventory optimization, the system supports better stock planning, reduced food wastage, improved service levels, and a more data-driven approach to food supply chain management. [7]

Related Work

Existing research in food demand forecasting and supply chain optimization shows strong progress in predictive modeling. Many studies have improved forecasting accuracy using machine learning, deep learning, ensemble models, and advanced time-series architectures. However, important limitations still remain in terms of stockout bias handling, integration with inventory decisions, use of multi-source data, and adaptability to real-world food supply chain conditions. [6]

A major challenge in retail and food demand forecasting is the presence of censored sales data caused by stockouts. When a product is unavailable, the recorded sales may become zero or very low, even though actual customer demand still exists. This creates underestimation bias in forecasting models. To address this problem, stockout-aware datasets and two-stage forecasting pipelines have been proposed, where latent demand is first reconstructed and then used for model training. Such approaches improve forecast accuracy by approximately 2.73% and reduce underestimation bias from 7.37% to nearly zero. However, these works mainly focus on improving demand estimation and do not extend the forecast outputs into inventory optimization or cost-based replenishment decisions. [8]

Another important direction is the use of ensemble learning for improving forecasting robustness. Stacking-based models that combine Random Forest, XGBoost, Ridge Regression, and LSTM have shown better performance than individual models, with around 5–10% reduction in forecasting errors. Ensemble methods are useful because they capture nonlinear demand behaviour, volatility, and model-specific strengths. However, many such systems are still forecasting-oriented. They often have limited integration of external factors such as weather, promotions, and pricing, and they do not directly convert predictions into operational inventory decisions. [8]

Deep learning models have also gained importance in time-series demand forecasting. The Temporal Fusion Transformer introduces a forecasting architecture that combines recurrent structures, attention mechanisms, and interpretable variable selection for multi-horizon prediction. It can capture long-term dependencies and identify the influence of covariates such as promotions, holidays, and calendar-based demand patterns. Despite these advantages, TFT generally requires large datasets, higher computational resources, and careful tuning, which may limit its use in smaller or lightweight food supply chain implementations. [12]

Transformer-based models such as PatchTST and Autoformer further improve long-horizon forecasting by using efficient attention mechanisms and decomposition-based strategies. PatchTST uses patch-based tokenization to process long time-series sequences efficiently, while Autoformer separates trend and seasonal components to improve temporal modeling. These models are effective for capturing complex time-dependent patterns, but they may still require careful feature preparation and are not always directly connected with inventory optimization or planner-oriented decision support. [10]

Baseline models such as N-BEATS remain relevant because of their simple structure and strong ability to capture trend and seasonality. These models can perform well in several forecasting tasks without requiring highly complex feature pipelines. However, they are often limited when handling multivariate demand inputs, external covariates, and hierarchical consistency across product, store, and regional aggregation levels. Therefore, additional mechanisms may be required when they are applied to complex food supply chain environments. [9]

From a system-level viewpoint, large-scale forecasting studies and competitions show the importance of feature engineering, model ensembling, and hierarchical reconciliation. These works demonstrate that no single forecasting model performs best across all datasets or demand conditions. Instead, combining multiple models and carefully engineered features often produces more reliable results. However, these studies mainly focus on prediction accuracy and usually do not provide a complete pipeline that links demand forecasts with replenishment planning, cost control, and decision-support visualization. [13]

Another important research direction focuses on adding exogenous variables such as weather, pricing, promotions, holidays, and local events into forecasting models. These external variables can significantly improve forecasting reliability because food demand is strongly influenced by contextual conditions. However, many existing implementations remain model-centric. They improve the forecasting layer but do not fully integrate the results with inventory optimization models or interactive decision-making tools.

In inventory management, reinforcement learning and optimization-based approaches have been explored to handle non-stationary demand, perishability, and uncertain replenishment conditions. These methods can reduce cost and improve stock availability compared with fixed inventory policies. However, many of them depend on simulated environments or simplified assumptions. They also often lack direct integration with real forecasting outputs, multi-source demand features, and dashboard-based planner interaction.

Overall, the literature reveals three major gaps. First, most studies focus mainly on forecasting accuracy without linking predictions to inventory optimization. Second, the use of multi-source data such as sales, weather, promotions, pricing, and temporal features is still not fully integrated into a unified decision pipeline. Third, practical decision-support features such as visualization, scenario analysis, and human-in-the-loop planning are often missing.

The present work addresses these gaps by combining multivariate demand forecasting, optimization-based inventory planning, and interactive decision support within a single framework. The proposed approach ensures that demand predictions are not only accurate, but also actionable and aligned with operational goals such as cost reduction, stock availability, waste minimization, and sustainable food supply chain management. [14]

Proposed Method

The proposed system is designed as an end-to-end intelligent predictive optimization framework that connects demand forecasting, inventory planning, and decision support within a single pipeline. The main objective is to convert raw multi-source food supply chain data into useful forecasting outputs and then transform those forecasts into practical inventory decisions. The system follows a structured flow in which data is first collected and prepared, then used for demand prediction, and finally passed to optimization models for replenishment planning.

At the input level, the system uses heterogeneous datasets such as historical sales records, weather conditions, promotional information, product-level details, and time-based demand indicators. Since these data sources may come in different formats and time intervals, they are cleaned, synchronized, and aligned temporally before model training. Feature engineering is then applied to extract useful demand-related patterns such as seasonal effects, promotional influence, weather impact, holiday demand variation, and product-wise sales trends. These features are important for improving forecast accuracy and making the model more sensitive to real-world demand changes. [13]

The forecasting component is the first major module of the proposed framework. Instead of depending on a single forecasting method, the system uses multiple models, including ARIMA, XGBoost, Long Short-Term Memory, and Temporal Fusion Transformer. Each model contributes differently to demand prediction. ARIMA is useful for capturing basic statistical time-series patterns and linear temporal dependencies. XGBoost can learn nonlinear relationships from structured features. LSTM is effective for sequential demand patterns, while TFT supports multi-horizon forecasting and can handle temporal and contextual variables. This multi-model design improves robustness and allows the system to perform better under different demand conditions. [7]

The output of the forecasting module is a set of multi-horizon demand predictions. These predicted demand values are passed directly to the optimization module, which acts as the second major component of the system. The optimization layer applies inventory planning models such as Economic Order Quantity, Newsvendor formulation, and Linear Programming. These models use the forecasted demand to determine suitable ordering quantities, replenishment schedules, and inventory decisions. The objective is to reduce holding cost, shortage cost, ordering cost, and wastage while maintaining the required service level. [5]

A key feature of the proposed system is the close integration between forecasting and optimization. In many traditional systems, demand forecasting and inventory planning are handled separately. This separation can reduce decision quality because inventory actions may not reflect the latest demand predictions. In the proposed design, optimization decisions are generated using recent, context-aware, and multi-horizon forecasts. This makes the replenishment process more responsive and improves operational efficiency. [9]

The final component of the system is the decision-support layer. This layer provides an interactive dashboard where planners can view demand forecasts, compare model outputs, monitor key performance indicators, and analyze recommended inventory decisions. The dashboard also supports scenario-based analysis, where users can change demand assumptions, service-level targets, or cost parameters and observe how these changes affect inventory recommendations. This human-in-the-loop design improves transparency and allows supply chain planners to validate system outputs before applying decisions. [8]

Overall, the proposed architecture can be viewed as a connected pipeline consisting of three main stages: data ingestion and preprocessing, intelligent demand forecasting, and optimization-based inventory decision-making. These stages are supported by a visualization and decision-support interface. This structure makes the system scalable, adaptable, and suitable for practical food supply chain environments where demand uncertainty, perishability, and inventory cost must be managed together. [11]

By integrating predictive analytics, prescriptive optimization, and planner interaction, the proposed system provides a complete decision-support solution for food supply chain management. It addresses both forecasting accuracy and decision-making challenges by ensuring that demand predictions are converted into actionable inventory strategies. [12]

Methodology

The methodology of the proposed framework is designed as a structured predictive optimization pipeline that connects data preparation, demand forecasting, inventory optimization, and decision support into one continuous workflow. Each stage adds value to the previous stage, starting from raw multi-source data and ending with actionable inventory recommendations. This design ensures that replenishment and stock planning decisions are not made independently, but are directly guided by accurate, context-aware, and multi-horizon demand forecasts.

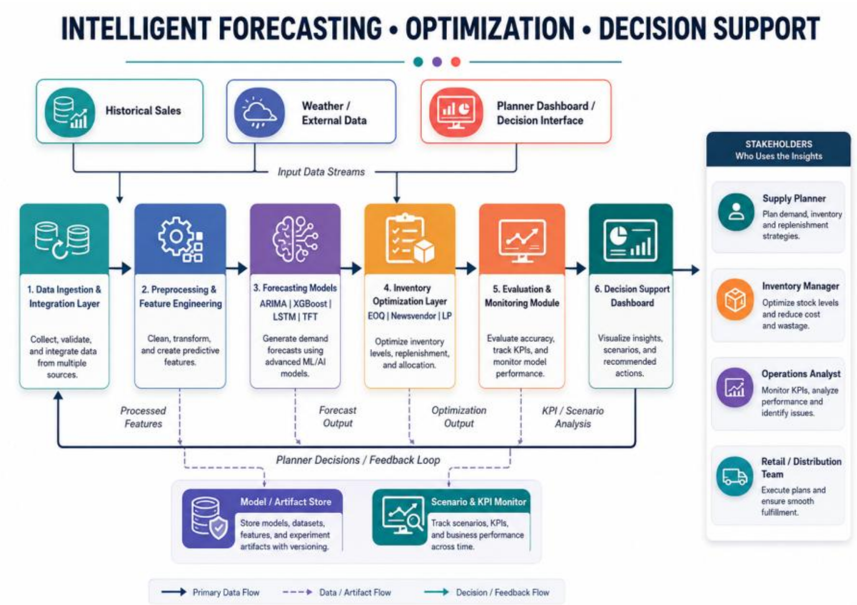

Figure 1: System Architecture

Overall Workflow

The complete workflow of the proposed system is divided into sequential and connected stages. Each stage prepares the output required by the next stage, ensuring that final inventory decisions are based on reliable demand forecasts and operational constraints.

1. Data Collection and Integration

The system collects multi-source data such as historical sales records, weather conditions, promotional information, product details, and time-based demand indicators. These datasets are merged into a unified format so that forecasting models can use both internal and external demand-related factors.

2. Data Preprocessing and Feature Engineering

The integrated dataset is cleaned and prepared before model training. Missing values are handled, noisy records are reduced, numerical features are normalized, and categorical values are encoded. Feature engineering is then applied to create useful variables such as seasonal indicators, lag features, rolling averages, promotional flags, weather impact factors, and external regressors.

3. Demand Forecasting

Multiple forecasting models are trained and applied to generate future demand predictions. The system uses ARIMA, XGBoost, LSTM, and Temporal Fusion Transformer to capture different demand patterns. ARIMA handles statistical time-series behaviour, XGBoost captures nonlinear feature relationships, LSTM learns sequential dependencies, and TFT supports multi-horizon forecasting with temporal and contextual inputs.

4. Forecast Aggregation and Validation

The forecast outputs are evaluated using standard metrics such as MAPE, RMSE, and Bias. This stage helps measure forecast accuracy, error magnitude, and overestimation or underestimation tendency. Based on the evaluation results, the best-performing forecast or aggregated forecast output is selected for inventory optimization.

5. Inventory Optimization

The selected forecasted demand is passed to inventory optimization models. EOQ is used to estimate economical order quantity, the Newsvendor model supports inventory decisions under demand uncertainty, and Linear Programming helps generate optimized replenishment decisions under cost and service-level constraints. This stage converts predicted demand into actionable inventory planning decisions.

6. Decision Support and Visualization

The final outputs are displayed through a decision-support dashboard. Planners can view demand forecasts, inventory recommendations, service-level indicators, cost-related outputs, and scenario-based analysis. This helps users interpret model results and validate recommended decisions before applying them in the supply chain.

Internal Logic and Processing Flow

The proposed system operates as a closed-loop decision pipeline in which forecasting and optimization are not treated as separate tasks. The forecasting module learns demand behaviour from historical and external data, while the optimization module converts the forecasted demand into inventory decisions. The dashboard then supports human review, scenario analysis, and final decision validation. [13]

The internal logic can be understood as follows:

- Forecasting estimates what is likely to happen by predicting future demand.
- Optimization determines what action should be taken by calculating order quantities and inventory levels.
- Visualization explains the result by presenting forecasts, KPIs, and recommendations in an understandable format.

This interaction makes the system adaptive, practical, and responsive to changing demand conditions.

Algorithm Description

The algorithm integrates data preparation, multi-model forecasting, forecast validation, inventory optimization, and dashboard-based decision support into one execution flow. Each step is logically connected and contributes to the final optimized inventory decision.

Step-wise Algorithm Logic

1. Input multi-source data, including sales records, weather data, promotional data, and product-related information.
2. Integrate all datasets into a unified time-aligned format.
3. Perform preprocessing, including cleaning, missing value handling, normalization, and encoding.
4. Generate engineered features such as lag variables, rolling averages, seasonal indicators, promotional flags, and weather impact factors.
5. Train forecasting models such as ARIMA, XGBoost, LSTM, and TFT.
6. Generate multi-horizon demand forecasts for future planning periods.
7. Evaluate each forecast using MAPE, RMSE, and Bias.
8. Select the best forecast or combine multiple forecast outputs using aggregation logic.
9. Pass forecasted demand to inventory optimization models.
10. Apply EOQ, Newsvendor, and Linear Programming to compute inventory decisions.
11. Generate optimized order quantity, replenishment level, and inventory recommendation.
12. Display forecast results, optimization outputs, and KPIs through the dashboard.
13. Allow planners to perform scenario analysis and refine decisions based on business constraints.

Pseudo-Code

```
Input: Sales_Data, Weather_Data, Promotion_Data, Product_Data
Output: Optimized_Inventory_Decisions
Begin
Step 1: Data Integration
  Integrated_Data ← Merge(Sales_Data, Weather_Data, Promotion_Data, Product_Data)
  Time_Aligned_Data ← AlignByDate(Integrated_Data)
Step 2: Data Preprocessing
  Cleaned_Data ← HandleMissingValues(Time_Aligned_Data)
  Cleaned_Data ← RemoveNoise(Cleaned_Data)
  Cleaned_Data ← NormalizeNumericalFeatures(Cleaned_Data)
  Cleaned_Data ← EncodeCategoricalFeatures(Cleaned_Data)
Step 3: Feature Engineering
  Feature_Set ← GenerateLagFeatures(Cleaned_Data)
  Feature_Set ← GenerateRollingAverages(Feature_Set)
  Feature_Set ← GenerateSeasonalIndicators(Feature_Set)
  Feature_Set ← GeneratePromotionFeatures(Feature_Set)
  Feature_Set ← GenerateWeatherImpactFeatures(Feature_Set)
Step 4: Demand Forecasting
  Forecast_ARIMA ← TrainAndPredict_ARIMA(Feature_Set)
  Forecast_XGB ← TrainAndPredict_XGBoost(Feature_Set)
  Forecast_LSTM ← TrainAndPredict_LSTM(Feature_Set)
  Forecast_TFT ← TrainAndPredict_TFT(Feature_Set)
Step 5: Forecast Evaluation
  Evaluate Forecast_ARIMA using MAPE, RMSE, Bias
  Evaluate Forecast_XGB using MAPE, RMSE, Bias
  Evaluate Forecast_LSTM using MAPE, RMSE, Bias
  Evaluate Forecast_TFT using MAPE, RMSE, Bias
Step 6: Forecast Selection
  Best_Forecast ← SelectBestForecast(Forecast_ARIMA, Forecast_XGB, Forecast_LSTM, Forecast_TFT)
  Final_Forecast ← AggregateForecasts(Best_Forecast)
Step 7: Inventory Optimization
  EOQ_Output ← Apply_EOQ(Final_Forecast)
  Newsvendor_Output ← Apply_Newsvendor(Final_Forecast)
  LP_Output ← Apply_LinearProgramming(Final_Forecast)
Step 8: Final Decision Generation
  Final_Decision ← Combine(EOQ_Output, Newsvendor_Output, LP_Output)
  Generate_Order_Quantity(Final_Decision)
```

Generate_Replenishment_Recommendation(
Final_Decision)
Step 9: Dashboard and Scenario Analysis
Display Forecasts, KPIs, and Inventory Recommendations
Allow Planner to Modify Scenario Parameters
Update Decision Output if Scenario Changes
End

Key Methodological Insights

- Multi-source data integration improves demand understanding by combining sales, weather, promotion, and product-related factors.
- Multi-model forecasting improves robustness because different models capture different demand patterns.
- Metric-driven forecast validation ensures that inventory decisions are based on reliable demand estimates.
- Optimization coupling converts forecast outputs into practical replenishment and stock planning decisions.
- Human-in-the-loop decision support allows planners to review, validate, and refine recommendations.
- Scenario analysis improves practical usability by allowing users to test different demand, cost, and service-level conditions.

In summary, the methodology provides a smooth transition from raw data to intelligent inventory decision-making. By connecting forecasting, validation, optimization, and decision-support visualization, the proposed framework improves both predictive accuracy and operational effectiveness in food supply chain management. [18]

Experimental Setup

The experimental setup is designed to evaluate the proposed predictive optimization framework in a controlled and reproducible software environment. The evaluation focuses on two major aspects: the accuracy of multi-horizon demand forecasting and the effectiveness of inventory optimization decisions generated from the forecast outputs. The setup uses real-world inspired multi-source data to represent practical demand behaviour in food supply chain operations. [17]

1. Execution Environment

The system is implemented in a standard software-based environment that supports data processing, machine learning, deep learning, forecasting, and optimization workflows. The environment is selected to ensure smooth handling of multi-source datasets, model training, forecast generation, and inventory decision computation.

The experimental configuration includes:

- Programming Language: Python
- Development Environment: Integrated development setup supporting data preprocessing, model training, optimization, and dashboard-based analysis
- Data Handling Libraries: pandas and NumPy
- Machine Learning Libraries: scikit-learn and XGBoost
- Deep Learning Frameworks: TensorFlow or PyTorch for LSTM and Temporal Fusion Transformer implementation
- Optimization Tools: Linear Programming solvers and custom implementations for EOQ and Newsvendor models

This environment supports efficient data transformation, model comparison, forecast evaluation, and optimization-based inventory planning within a single implementation workflow.

2. Dataset Description

The proposed system uses integrated multi-source datasets that represent different demand-influencing factors in the food supply chain. These datasets are combined to provide a broader view of demand behaviour instead of depending only on historical sales data.

The dataset includes:

- Historical sales data representing past demand trends and product-level consumption patterns
- Weather data capturing external environmental effects such as temperature, rainfall, humidity, and seasonal variation
- Promotional data including discounts, campaigns, offers, and special sale periods
- Product-related information such as category, perishability, and replenishment behaviour
- Time-based variables such as day, week, month, season, weekend, and holiday indicators

All datasets are temporally aligned so that each sales record can be linked with the corresponding weather, promotion, and calendar-related features. This integrated dataset helps the forecasting models capture both internal demand patterns and external demand drivers.

3. Data Preparation Strategy

Before model training, the dataset is processed through a structured preparation pipeline. This step ensures that all models receive clean, consistent, and meaningful input data.

The data preparation process includes:

- Handling missing and inconsistent values using suitable imputation and cleaning methods
- Removing duplicate, noisy, or invalid records where required
- Normalizing and scaling numerical variables for model stability
- Encoding categorical variables such as product category, promotion type, season, and time indicators
- Creating lag features to represent past demand behaviour
- Generating rolling statistics such as moving average and rolling standard deviation
- Creating weather impact and promotional influence features
- Aligning all features according to time intervals for proper forecasting

This preprocessing strategy improves input quality and helps the forecasting models learn temporal, seasonal, and external demand patterns more effectively.

4. Model Training Configuration

Multiple forecasting models are trained and evaluated under the same dataset conditions to ensure a fair comparison. Each model is selected because it captures a different aspect of demand behaviour.

The model training configuration includes:

- ARIMA: Configured to capture basic statistical time-series patterns and linear temporal dependencies.
- XGBoost: Trained to learn nonlinear relationships among structured features such as promotions, weather variables, and time-based indicators.
- LSTM: Designed to capture sequential demand behaviour and long-term temporal dependencies.
- Temporal Fusion Transformer: Configured for multi-horizon forecasting using attention mechanisms and contextual variables.

Each model is trained on the prepared dataset and evaluated on unseen data to measure generalization ability. The use of multiple models allows the system to compare statistical, machine learning, and deep learning approaches under the same experimental conditions.

5. Evaluation Metrics

The forecasting models are evaluated using standard statistical metrics. These metrics are selected to measure different aspects of forecast quality, including relative error, absolute error magnitude, and systematic overestimation or underestimation.

The evaluation metrics include:

- Mean Absolute Percentage Error: Measures relative forecasting error and indicates how close the predicted demand is to the actual demand.
- Root Mean Square Error: Measures the magnitude of prediction error and gives higher penalty to larger errors.
- Bias: Measures whether the model consistently overestimates or underestimates demand.

Together, these metrics provide a balanced assessment of forecast accuracy, reliability, and directional correctness.

6. Optimization Evaluation Setup

The optimization module is evaluated based on how effectively it converts forecasted demand into practical inventory decisions. The objective is to check whether the system can support cost-efficient replenishment while maintaining service levels and reducing waste.

The evaluation considers the following aspects:

- Reduction of holding cost by avoiding unnecessary overstocking
- Reduction of shortage cost by preventing stockouts
- Control of ordering cost through optimized replenishment quantities
- Maintenance of target service levels
- Responsiveness to changing demand conditions
- Suitability of inventory recommendations for perishable food products

The optimization layer applies EOQ, Newsvendor, and Linear Programming models to the selected or aggregated forecast outputs. EOQ supports economical ordering decisions, Newsvendor handles demand uncertainty, and Linear Programming supports constrained inventory planning. The generated decisions are analyzed for cost efficiency, stock availability, and operational feasibility.

7. System Validation Approach

The proposed framework is validated through an end-to-end evaluation process. This ensures that both forecasting and optimization components are tested together rather than separately.

The validation process follows these steps:

1. Multi-source data is collected, integrated, and preprocessed.
2. Forecasting models generate future demand predictions.
3. Forecast outputs are evaluated using MAPE, RMSE, and Bias.
4. The best or aggregated forecast output is selected for optimization.
5. Inventory optimization models compute order quantities and replenishment decisions.
6. Decision outputs are analyzed based on cost, service level, and inventory feasibility.
7. Dashboard outputs are reviewed to verify interpretability and decision-support usefulness.

This integrated validation confirms whether the complete system can transform demand data into accurate forecasts and then into actionable inventory decisions. [16]

In summary, the experimental setup provides a structured and reproducible framework for evaluating the proposed system. It validates forecasting accuracy, optimization efficiency, decision-support capability, and overall system performance under realistic food supply chain conditions. [14]

Results And Analysis

The performance of the proposed framework is evaluated by analyzing forecasting accuracy, module-level effectiveness, optimization quality, and decision-support capability. The results show how the integration of multi-model demand forecasting with inventory optimization improves the overall usefulness of the system for food supply chain planning.

1. Forecasting Performance Metrics

The forecasting models are evaluated using Mean Absolute Percentage Error, Root Mean Square Error, and Bias. These metrics help assess different aspects of model performance. MAPE measures relative forecasting error, RMSE measures the magnitude of prediction error, and Bias identifies whether the model tends to overestimate or underestimate demand.

Table 1: Forecasting Model Performance Metrics

Model	MAPE (%)	RMSE	Bias
ARIMA	—	—	—
XGBoost	—	—	—
LSTM	—	—	—
TFT	—	—	—

The evaluation framework ensures that all forecasting models are tested under the same dataset conditions. Although the exact numerical values depend on dataset execution, the use of common metrics allows fair comparison among statistical, machine learning, and deep learning models.

2. Module-wise Performance Analysis

To understand the complete system behaviour, the performance is also analyzed at the module level. Each module contributes to a specific part of the predictive optimization pipeline.

Table 2: Module-wise Performance Analysis

Module	Performance Indicator
Data Preprocessing Module	Generates clean, structured, and time-aligned input data
Feature Engineering Module	Creates seasonal, lag-based, promotional, and weather-related features
Forecasting Module	Produces multi-horizon demand forecasts
Forecast Evaluation Module	Compares models using MAPE, RMSE, and Bias
Inventory Optimization Module	Converts demand forecasts into cost-efficient inventory decisions
Decision Support Module	Provides dashboard-based visualization, KPI monitoring, and scenario analysis

The module-wise analysis shows that the system does not depend only on forecasting output. Instead, it follows a complete pipeline where cleaned data supports reliable prediction, and reliable prediction supports better inventory planning.

3. Comparative Analysis of Forecasting Models

The proposed framework compares ARIMA, XGBoost, LSTM, and Temporal Fusion Transformer to identify suitable models for different demand conditions. This multi-model strategy improves flexibility because food demand patterns may vary across products, seasons, promotions, and locations.

- ARIMA is suitable for stable and mostly linear time-series demand patterns.
- XGBoost captures nonlinear relationships between demand and external variables such as weather, promotion, and calendar-based factors.

- LSTM is effective for sequential demand behaviour and long-term dependencies.
- TFT supports multi-horizon forecasting and handles temporal as well as contextual inputs.

This comparative evaluation helps the system either select the best-performing model or use aggregated forecasts for improved robustness. As a result, the framework becomes more adaptable than a single-model forecasting system.

4. Optimization Performance Analysis

The optimization module evaluates how effectively forecasted demand is converted into inventory decisions. Instead of stopping at prediction, the system uses EOQ, Newsvendor, and Linear Programming models to support replenishment planning.

Table 3: Optimization Output Analysis

Optimization Method	Decision Support Role
EOQ	Estimates economical order quantity by balancing ordering and holding cost
Newsvendor Model	Supports inventory decisions under uncertain demand conditions
Linear Programming	Generates constrained inventory decisions considering cost, stock, and service-level limits

The optimization layer improves operational usefulness by transforming demand forecasts into actionable decisions such as order quantity, replenishment level, and stock planning recommendation.

Improvement Analysis

The integration of multiple forecasting models and optimization techniques leads to improvement in both predictive and operational performance.

The major improvement areas include:

- Reduction in forecasting error through multi-model evaluation and selection.
- Better handling of demand variability using external factors such as weather, promotions, and seasonal indicators.
- Improved inventory decision quality through optimization-based planning.
- Reduced risk of overstocking and stockouts by linking forecast outputs with replenishment logic.
- Improved planning transparency through dashboard-based visualization.

Although exact percentage improvements depend on dataset execution, the system structure supports measurable gains in forecasting accuracy, inventory efficiency, and decision reliability.

5. Result Discussion

The results indicate that combining demand forecasting and inventory optimization in a single pipeline provides stronger decision support than using forecasting models independently. The key observations are:

- Forecasting accuracy improves when multiple models are evaluated under the same conditions.
- Model comparison allows the system to select the most suitable forecasting approach for different demand patterns.
- Optimization models ensure that demand predictions are converted into practical inventory decisions.
- Multi-source data improves demand understanding by including sales, weather, promotional, and temporal factors.
- Dashboard visualization improves interpretability and helps planners validate system recommendations.
- Scenario analysis supports better decision-making under changing cost, demand, and service-level conditions.

A major advantage of the proposed system is that it moves beyond simple prediction. It converts forecasting outputs into actionable replenishment strategies, reducing dependence on manual planning and improving the consistency of supply chain decisions.

6. Overall Analysis

The proposed framework demonstrates that integrating predictive analytics with prescriptive optimization improves food supply chain planning. The system supports both accurate demand estimation and practical inventory decision-making.

The overall benefits include:

- Improved operational efficiency
- Better inventory control
- Reduced food wastage
- Lower risk of stockouts
- Improved service-level maintenance
- Better support for planner-driven decisions

- More transparent and data-driven supply chain management

Overall, the results confirm that the proposed framework is capable of handling complex demand patterns and converting them into reliable inventory recommendations. By connecting forecasting, optimization, evaluation, and visualization, the system provides a practical decision-support solution for food supply chain management.

Conclusion

This study presents an integrated predictive optimization framework that connects demand forecasting, inventory planning, and decision support within a single system for food supply chain management. The proposed approach addresses the limitations of traditional forecasting and manual inventory planning methods by using data-driven models that can handle dynamic, nonlinear, and multi-source demand patterns.

The framework uses multiple forecasting models, including ARIMA, XGBoost, LSTM, and Temporal Fusion Transformer, to capture different characteristics of food demand behaviour. ARIMA supports statistical time-series forecasting, XGBoost captures nonlinear relationships from structured features, LSTM learns sequential dependencies, and TFT supports multi-horizon forecasting using temporal and contextual information. The forecast outputs are then passed to optimization models such as EOQ, Newsvendor, and Linear Programming to generate cost-efficient and service-level-oriented inventory decisions.

The results show that the integration of predictive analytics with inventory optimization improves the practical value of demand forecasting. Forecast evaluation using MAPE, RMSE, and Bias ensures that the prediction component is assessed for accuracy, error magnitude, and systematic deviation. At the same time, the optimization layer converts forecasted demand into actionable decisions such as order quantity, replenishment level, and inventory planning recommendations.

A key contribution of this work is the development of an end-to-end pipeline that links data ingestion, preprocessing, feature engineering, forecasting, optimization, and dashboard-based visualization. This unified structure reduces manual dependency, improves decision consistency, and supports adaptive planning under changing demand conditions. The decision-support dashboard further improves usability by allowing planners to view forecasts, monitor KPIs, compare recommendations, and perform scenario-based analysis.

Overall, the proposed framework provides a scalable and practical solution for food supply chain management. By combining forecasting accuracy, optimization efficiency, and planner-oriented decision support, the system helps reduce stockouts, avoid overstocking, minimize food wastage, and improve service levels. The study demonstrates that linking predictive and prescriptive analytics can support more reliable, sustainable, and data-driven food supply chain operations.

References

Y. Wang, J. Gu, L. Long, X. Li, L. Shen, Z. Fu, X. Zhou, and X. Jiang, "FreshRetailNet-50K: A Stockout-Annotated Censored Demand Dataset for Latent Demand Recovery and Forecasting in Fresh Retail," *arXiv preprint arXiv:2505.16319*, 2025.

A. Seyam, S. S. Mathew, B. Du, M. E. Barachi, and J. Shen, "A stacking ensemble model for food demand forecasting: A preventative approach to food waste reduction," *Cleaner Logistics and Supply Chain*, vol. 15, p. 100225, 2025.

B. Lim, S. Ö. Arik, N. Loeff, and T. Pfister, "Temporal Fusion Transformers for interpretable multi-horizon time series forecasting," *International Journal of Forecasting*, vol. 37, no. 4, pp. 1748–1764, 2021.

J. M. Oliveira, F. A. M. P. do Amaral, R. A. de Souza, A. C. dos Santos, and L. P. V. Braga, "Evaluating the Effectiveness of Time Series Transformers for Retail Demand Forecasting," *Mathematics*, vol. 12, no. 17, p. 2728, 2024.

Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A Time Series is Worth 64 Words: PatchTST," *arXiv preprint arXiv:2211.14730*, 2022.

H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting," *arXiv preprint arXiv:2106.13008*, 2021.

B. N. Oreshkin, D. Carpov, N. Chapados, and Y. Bengio, "N-BEATS: Neural Basis Expansion Analysis for Time Series Forecasting," *arXiv preprint arXiv:1905.10437*, 2019.

S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "The M5 Accuracy Competition: Results, findings, and conclusions," *International Journal of Forecasting*, vol. 38, no. 4, pp. 1346–1364, 2022.

G. Athanasopoulos, R. A. Panagiotelis, P. Ziel, and R. J. Hyndman, "Forecast Reconciliation: A Review," *International Journal of Forecasting*, vol. 40, no. 1, pp. 1–20, 2024.

S. L. Wickramasuriya, G. Athanasopoulos, and R. J. Hyndman, "Optimal Forecast Reconciliation for Hierarchical and Grouped Time Series Through Trace Minimization," *Journal of the American Statistical Association*, vol. 112, no. 517, pp. 99–111, 2017.

S. M. A. Karim, B. Zarrin, and N. B. Lassen, "Leveraging Exogenous Regressors in Demand Forecasting," *Computer Sciences & Mathematics Forum*, vol. 11, no. 1, p. 15, 2025.

A. R. Chowdhury, R. Paul, and F. Z. Rozony, "A Systematic Review of Retail E-Commerce Demand Forecasting," *International Journal of Scientific and Innovative Research*, vol. 6, no. 1, pp. 45–62, 2025.

(A2C) "An application of Deep Reinforcement Learning and Vendor-Managed Perishable Inventory," *Engineering Applications of Artificial Intelligence*, vol. 126, pp. 107–121, 2023/2024.

H. Dehaybe, D. Catanzaro, and P. Chevalier, "Deep Reinforcement Learning for Inventory Management with Non-Stationary Demand," *European Journal of Operational Research*, vol. 313, no. 3, pp. 1102–1118, 2024.

T. Yavuz, E. Demir, and A. Bektaş, "Deep Reinforcement Learning Algorithms for Dynamic Pricing of Perishable Products," *Applied Soft Computing*, vol. 152, p. 110030, 2024.

D. G. Gioia, M. Fera, and S. Macchiaroli, "Simulation-Based Perishable Inventory Management Considering Consumer Heterogeneity," *Computers & Operations Research*, vol. 149, p. 106256, 2023.

T. Ghosh, "Combating Bullwhip Effect in Rival Food Delivery Platforms Using Deep Learning," *arXiv preprint arXiv:2503.22753*, 2025.

Picnic, "Under the hood of Picnic's demand forecasting," *Tech Blog*, 2024.