



ARTICLE

Hybrid Deep Learning Approach for Coupled Demand Forecasting and Supply Chain Optimization

Nusrat Yasmin Nadia¹, Md Habibul Arif², Habibor Rahman Rabby³, Md Iftekhar Monzur Tanvir¹,
Md Jakir Hossen^{4,*} and M. F. Mridha⁵

¹Department of Information Technology, Washington University of Science and Technology, Alexandria, VA, USA

²Department of Information Technology, University of the Potomac, Washington, DC, USA

³Department of Computer Science, Campbellsville University, Louisville, KY, USA

⁴Center for Advanced Analytics (CAA), COE for Artificial Intelligence, Faculty of Engineering & Technology (FET), Multimedia University, Melaka, Malaysia

⁵Department of Computer Science and Engineering, American International University-Bangladesh (AIUB), Dhaka, Bangladesh

*Corresponding Author: Md Jakir Hossen. Email: jakir.hossen@mmu.edu.my

Received: 06 October 2025; Accepted: 26 February 2026; Published: 23 July 2026

ABSTRACT: Supply chain resilience and efficiency are vital in industries characterized by volatile demand and uncertain supply, such as textiles and personal protective equipment (PPE). Traditional forecasting and optimization approaches often operate in isolation, limiting their real-world effectiveness. This paper proposes a Hybrid AI Framework for Demand–Supply Forecasting and Optimization (HAF-DS), which integrates a Long Short-Term Memory (LSTM)–based demand forecasting module with a mixed-integer linear programming (MILP) optimization layer. The LSTM captures temporal and contextual demand dependencies, while the optimization layer prescribes cost-efficient replenishment and allocation decisions. The framework jointly minimizes forecasting error and operational cost through embedding-based feature representation and recurrent neural architectures. Experiments on textile sales and supply chain datasets show significant performance gains over statistical and deep learning baselines. On the combined dataset, HAF-DS reduced Mean Absolute Error (MAE) from 15.04 to 12.83 (14.7%), Root Mean Squared Error (RMSE) from 19.53 to 17.11 (12.4%), and Mean Absolute Percentage Error (MAPE) from 9.5% to 8.1%. Inventory cost decreased by 5.4%, stockouts by 27.5%, and service level rose from 95.5% to 97.8%.

KEYWORDS: AI-driven supply chain; demand forecasting; supply chain optimization; deep learning; textile industry; PPE manufacturing; hybrid framework

1 Introduction

The global landscape of supply chain management has become increasingly complex due to rapid globalization, shorter product life cycles, and heightened demand volatility. Industries such as textiles and personal protective equipment (PPE) face challenges from fluctuating consumer demand, crises such as pandemics, and disruptions in logistics and supplier networks [1]. Recent studies emphasize digital transformation and hybrid AI for real-time adaptation and resilience [2].

The main research problem addressed here is the disconnect between demand forecasting and supply optimization. Conventional forecasting models including statistical time-series and machine learning methods focus on prediction accuracy but do not convert forecasts into actionable supply chain policies. Likewise, optimization models often rely on simplifying assumptions that fail to capture real-world variability, making

their recommendations less reliable [3]. This gap between predictive analytics and operational decision-making is critical in textile and PPE supply chains, where inaccurate forecasts or inefficient replenishment lead to overstocking, shortages, and service-level issues.

This study aims to design and evaluate a hybrid AI framework that unifies forecasting and optimization into an integrated system. The objective is not only to improve forecast accuracy but also to ensure that predictive insights drive cost-efficient and service-oriented decisions. The proposed framework handles large-scale data, integrates contextual variables such as inventory levels, lead times, and supplier variability, and delivers actionable outcomes that enhance both accuracy and efficiency.

The research holds broader importance beyond methodological innovation. In textile and PPE supply chains, accurate forecasting and effective optimization are essential not only for cost efficiency but also for social and economic resilience. For instance, during global health crises, PPE shortages underscore the importance of robust supply chains. By combining deep learning-based forecasting with optimization, this approach addresses both predictive and prescriptive needs, improving decision quality, reducing cost volatility, and enhancing service levels.

Unlike conventional hybrid or two-stage supply chain models that treat demand forecasting and supply optimization as sequential and loosely coupled processes, the proposed framework introduces a fully integrated predictive-prescriptive learning architecture. The key novelty lies in the end-to-end, differentiable coupling between the forecasting and optimization modules, where supply-side operational objectives directly influence the learning of the demand forecasting model through shared gradients. Specifically, the proposed Hybrid AI Framework for Demand-Supply Forecasting and Optimization (HAF-DS) jointly minimizes forecasting error and operational cost using a unified loss function, allowing downstream cost, service-level, and constraint violations to actively guide the update of forecasting parameters during training. This design ensures that improvements in predictive accuracy consistently translate into tangible operational benefits, addressing a critical limitation of traditional predict-then-optimize and two-stage hybrid approaches.

To achieve these objectives, the study develops a deep learning-based hybrid framework that integrates recurrent neural architectures with supply-side optimization modules. The methodology involves preprocessing heterogeneous datasets from textile sales and supply chain operations, applying embedding-based feature representations, and jointly training forecasting and optimization components. The model is evaluated using mean absolute error (MAE), root mean squared error (RMSE), mean absolute percentage error (MAPE), inventory cost, stockout rate, service level, and total cost. The key contributions are summarized below:

- A novel hybrid AI framework that jointly optimizes demand forecasting and supply chain operations, bridging a critical gap in current research.
- A data preprocessing pipeline leveraging embeddings and data augmentation for enhanced feature representation.
- Comprehensive experiments on textile and PPE datasets showing superior performance over classical and deep learning baselines.
- An ablation study highlighting the contribution of key components such as embeddings and joint training.
- Practical insights into the deployment of integrated predictive-prescriptive systems under volatile demand and uncertain supply.

The remainder of this paper is structured as follows. [Section 2](#) reviews related work on forecasting and optimization in supply chains. [Section 3](#) presents the proposed methodology, including preprocessing,

architecture, and training. [Section 4](#) reports results and ablation analyses. [Section 5](#) discusses implications, limitations, and future directions, while [Section 6](#) concludes the study.

2 Related Work

Research on supply chain forecasting and optimization has evolved along two largely independent but complementary trajectories: demand forecasting using statistical and machine learning techniques, and supply chain optimization using heuristic and mathematical programming approaches. Traditional forecasting methods such as ARIMA, exponential smoothing, and regression models have long served as baselines for predicting demand trends in manufacturing and retail [4]. While these methods are interpretable and effective for short-term prediction in stable environments, they often struggle with nonlinearities, seasonality, and abrupt demand shifts, which are characteristic of dynamic markets such as textile and PPE supply chains.

To address these limitations, more recent studies have employed machine learning models, including support vector regression, random forests, and gradient boosting, which are better suited for capturing nonlinear dependencies. However, these approaches still rely heavily on manual feature engineering and often fail to fully exploit temporal dependencies inherent in demand time series.

The advent of deep learning has led to significant progress in time-series demand forecasting. Models such as recurrent neural networks (RNNs), long short-term memory (LSTM) networks, and gated recurrent units (GRUs) have been widely adopted to capture sequential dependencies and long-term temporal patterns [5]. These architectures have been applied across multiple industries and consistently demonstrate improved predictive accuracy over traditional statistical and machine learning approaches.

More recent advancements incorporate attention mechanisms and transformer-based models, which further enhance the ability to model long-range dependencies and heterogeneous feature interactions. Despite these improvements, many deep learning approaches remain focused primarily on forecasting accuracy, with limited direct integration into operational decision-making processes [6–10].

Recent benchmarks confirm that Transformer-based architectures outperform recurrent models in time-series forecasting, offering improved generalization and scalability [11]. These models enable data-driven adaptation in volatile markets, making them suitable for high-variance domains such as PPE and textile manufacturing. Recent empirical benchmarking in logistics also shows that the relative ranking of statistical, tree-based, and deep models depends on noise and demand complexity, motivating the use of both empirical evidence and theory-driven design choices in hybrid frameworks [12].

Parallel to the forecasting literature, supply chain optimization has traditionally relied on operations research methods such as linear programming, mixed-integer programming, and stochastic optimization [13]. These methods have been applied to problems including inventory control, production planning, transportation scheduling, and supplier selection. Classical optimization frameworks, such as the economic order quantity (EOQ) model and reorder point heuristics, remain foundational tools in practice. Although these models provide tractable solutions, they often rely on simplifying assumptions, such as stationary demand or deterministic lead times, which limit their applicability in volatile and uncertain environments. Extensions that incorporate stochastic elements or robust optimization frameworks have improved realism, but scalability and adaptability to high-dimensional, real-world supply chains remain challenges.

More recently, research has turned toward integrating machine learning and optimization to create predictive–prescriptive systems. These hybrid approaches leverage demand forecasts as inputs to optimization models, enabling more accurate and adaptive decision-making [14]. Examples include coupling

LSTM forecasts with stochastic inventory control policies or embedding neural forecasts within simulation-based optimization. Reinforcement learning has also emerged as a promising direction, where agents learn policies for inventory and replenishment decisions through interaction with simulated supply chain environments. These methods move beyond static optimization toward adaptive decision-making but often remain constrained by computational complexity and limited generalizability across industries [15,16].

From a theoretical perspective, decision-focused learning and predict-then-optimize analysis formalize how learning objectives should align with downstream operational costs rather than forecast error alone [17,18]. Complementary empirical studies evaluate these ideas on inventory and replenishment benchmarks and report improved cost–service trade-offs under lost-sales and multi-echelon settings [19,20].

Recent work has increasingly focused on strengthening the link between demand forecasting and operational decision-making in supply chain systems. For example, recent studies integrate deep learning–based demand prediction with inventory or replenishment optimization to improve robustness under uncertainty. Other contributions emphasize hybrid machine learning and optimization frameworks to enhance supply chain resilience and cost efficiency in volatile environments. While these approaches demonstrate improved coordination between prediction and decision stages, most of them still adopt a two-stage or sequential design in which forecasting and optimization are trained separately. In contrast, the proposed HAF-DS framework enables joint, end-to-end learning of forecasting and supply-side objectives, allowing operational cost and service-level feedback to directly influence predictive model training.

Contemporary studies have expanded on hybrid predictive–prescriptive designs that integrate reinforcement learning, differentiable optimization, and deep forecasting [21]. These works highlight the importance of bridging predictive analytics and operational optimization for real-time decision support, motivating the unified design adopted in this study.

In parallel, recent advancements in digital twin (DT) frameworks have introduced virtual replicas of supply chain systems that enable real-time monitoring, simulation, and predictive control [22,23]. These digital twins integrate sensor data, forecasting modules, and optimization engines to create adaptive, data-driven decision environments. However, most existing DT implementations focus on descriptive and predictive analytics, while prescriptive optimization remains loosely coupled or rule-based. By embedding differentiable optimization directly within a learning framework, our proposed hybrid architecture can serve as a foundation for AI-enhanced digital twins, where both prediction and decision components co-evolve through joint training. This unified design positions HAF-DS as a next-generation framework that bridges data-driven forecasting, decision optimization, and DT-based supply chain management.

Overall, prior research has made important contributions to demand forecasting and supply chain optimization, but largely in isolation. Forecasting studies have focused on improving prediction accuracy through statistical, machine learning, and deep learning models, while optimization research has concentrated on cost-efficient inventory, production, and replenishment policies under predefined assumptions.

However, most existing approaches treat forecasting and optimization as sequential or loosely connected processes [24,25]. Forecasting models are typically trained to minimize prediction error without considering downstream operational consequences, whereas optimization models often rely on externally provided forecasts and simplified constraints that limit adaptability under volatile demand and uncertain supply conditions. As a result, prediction improvements do not always translate into operational efficiency.

In contrast, the proposed Hybrid AI Framework for Demand–Supply Forecasting and Optimization (HAF-DS) introduces a unified predictive–prescriptive architecture in which demand forecasting and supply-side optimization are jointly trained within a single learning framework. By enabling optimization objectives such as cost and service level to directly influence forecasting parameters, and by incorporating

embedding-based contextual representations, the proposed approach achieves coordinated improvement in both predictive accuracy and operational decision quality. This end-to-end joint optimization constitutes the core novelty of the proposed work.

3 Methodology

This section presents the proposed hybrid AI framework for demand–supply forecasting and optimization in textile and PPE supply chains. The methodology integrates deep learning-based demand forecasting with supply-side constraint modeling and optimization. The subsections cover model architecture, forecasting module, supply chain optimization module, hybrid integration, and training strategy.

Unlike existing hybrid models that integrate forecasting and optimization in a sequential manner, the proposed framework is trained end-to-end using a shared, differentiable objective that enables direct gradient flow from supply-side optimization losses to the forecasting network. The proposed framework is theoretically grounded in predictive–prescriptive modeling, combining statistical learning and mathematical optimization. Specifically, the forecasting module learns a function f_θ that captures temporal demand dependencies, while the optimization module defines a constrained decision function g_ϕ that minimizes total operational cost given the predicted demand. The combined objective function:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{forecast} + \lambda_2 \mathcal{L}_{supply} \quad (1)$$

establishes a unified mathematical formulation that links prediction accuracy to decision efficiency. This formulation aligns with hybrid AI theory, ensuring that the model adheres to both data-driven learning principles and optimization-based reasoning. By introducing this theoretical foundation, the methodology transitions from descriptive system design to a formal predictive–prescriptive framework.

3.1 Analytical Formulation and Gradient Coupling

The key analytical contribution of the proposed hybrid AI framework lies in the differentiable coupling between the predictive (forecasting) and prescriptive (optimization) modules. Unlike traditional two-stage systems that optimize decisions after prediction, our model establishes an end-to-end learning process where the optimization loss directly influences the forecasting parameters through shared gradients.

Formally, the forecasting network learns a mapping $f_\theta : \mathbf{x}_t \rightarrow \hat{y}_{t+1}$, while the optimization module learns a mapping $g_\phi : (\hat{y}_{t+1}, \mathbf{s}_t) \rightarrow \mathbf{o}_{t+1}$ that minimizes the supply-side loss $\mathcal{L}_{supply}$. During joint training, the total loss is given by:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{forecast} + \lambda_2 \mathcal{L}_{supply} \quad (2)$$

The gradient-based parameter updates are computed as:

$$\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{total}, \quad \phi \leftarrow \phi - \eta \nabla_\phi (\lambda_2 \mathcal{L}_{supply}) \quad (3)$$

This coupling ensures that forecasting parameters θ are indirectly optimized with respect to downstream cost efficiency, creating a feedback loop between predictive accuracy and operational performance. Conceptually, this joint optimization transforms the framework from a sequential pipeline into a unified mathematical model that aligns learning objectives across both modules.

To account for uncertainty in demand prediction, we incorporate stochastic regularization by sampling predicted demand from a Gaussian distribution $\hat{y}_{t+1} \sim \mathcal{N}(\mu_\theta, \sigma_\theta^2)$, allowing the optimization module to adapt

to confidence-weighted forecasts. This probabilistic gradient coupling further enhances decision robustness under volatile demand and supply fluctuations.

This analytical coupling defines the mathematical backbone of the proposed framework, establishing how predictive gradients inform prescriptive optimization in a fully differentiable learning process.

3.2 Data Preprocessing

To ensure the datasets were suitable for deep learning models, a systematic preprocessing pipeline was applied to both *Textile Sales Data* and *Supply Chain Data*. The preprocessing stage included handling missing values, normalization, categorical encoding, time-series sequence generation, and data partitioning into training, validation, and testing subsets.

3.2.1 Data Cleaning and Missing Values

Both datasets contained instances of missing values due to incomplete transaction logs or unavailable supply chain attributes. We used mean imputation for continuous attributes and mode imputation for categorical attributes. Formally, for a continuous feature x_j , the imputed value $\hat{x}_{i,j}$ for sample i is defined as:

$$\hat{x}_{i,j} = \begin{cases} x_{i,j}, & \text{if } x_{i,j} \neq \text{NaN} \\ \mu_j, & \text{if } x_{i,j} = \text{NaN} \end{cases} \quad (4)$$

where $\mu_j = \frac{1}{N} \sum_{i=1}^N x_{i,j}$ is the mean of feature j . For categorical features, the most frequent category was used.

3.2.2 Feature Normalization

Since deep neural networks are sensitive to feature scales, we applied Min–Max normalization to continuous variables such as quantities, values, stock levels, and lead times. Each feature x_j was scaled into the range $[0, 1]$ using:

$$x'_{i,j} = \frac{x_{i,j} - \min(x_j)}{\max(x_j) - \min(x_j)} \quad (5)$$

This ensured that all numerical features contributed equally to the learning process and improved gradient stability during training.

3.2.3 Categorical Encoding

Categorical variables such as *Product Type*, *SKU*, and *Supplier Name* were transformed using embedding representations. First, categorical values were indexed and converted into one-hot vectors. Subsequently, we employed a trainable embedding matrix $E \in \mathbb{R}^{|C| \times d}$, where $|C|$ is the number of unique categories and d is the embedding dimension. For a categorical input c_k , the embedded representation is:

$$\mathbf{z}_k = E^\top \mathbf{1}_{c_k} \quad (6)$$

where $\mathbf{1}_{c_k}$ is a one-hot vector of category c_k . This enabled the network to learn distributed semantic representations of categorical features.

3.2.4 Time-Series Sequence Preparation

For demand forecasting, sales data were transformed into supervised learning sequences. Given a time series $\{y_t\}_{t=1}^T$ representing demand (quantities sold), we generated input windows of length L and predicted the next step y_{t+1} . Formally:

$$\mathbf{x}_t = [y_{t-L+1}, y_{t-L+2}, \dots, y_t], \quad \hat{y}_{t+1} = f(\mathbf{x}_t) \quad (7)$$

where $f(\cdot)$ denotes the deep learning forecasting model. This sliding-window approach converted raw sales data into structured input-output pairs suitable for recurrent and transformer-based architectures.

3.2.5 Train–Validation–Test Split

The datasets were partitioned into training, validation, and testing sets to ensure unbiased evaluation. For time-series forecasting, we used chronological splits:

$$\mathcal{D}_{train} = \{y_t \mid t = 1, \dots, T_{train}\}, \quad \mathcal{D}_{val} = \{y_t \mid t = T_{train} + 1, \dots, T_{val}\}, \quad \mathcal{D}_{test} = \{y_t \mid t = T_{val} + 1, \dots, T\} \quad (8)$$

with 70% of the data allocated to training, 15% to validation, and 15% to testing. This partitioning strategy ensured that the validation set guided hyperparameter tuning while the testing set provided an unbiased measure of generalization performance.

3.2.6 Data Augmentation for Robustness

To improve generalization under demand fluctuations and supply uncertainties, we applied data augmentation strategies such as Gaussian noise injection into sales values:

$$y'_t = y_t + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2) \quad (9)$$

and random dropout of supplier records to simulate disruptions. These augmentations enhanced the robustness of the model against real-world stochastic variations.

3.2.7 Final Dataset

After preprocessing, the textile sales dataset yielded time-series sequences for demand prediction, while the supply chain dataset provided contextual features (e.g., stock levels, lead times, defect rates). The unified preprocessed dataset was used as input to the hybrid AI framework for integrated demand–supply forecasting and optimization.

To enhance contextual representation, all categorical and temporal features were transformed using embedding layers that map discrete variables (e.g., supplier ID, product category, seasonality index) into dense vector spaces. This allows the model to capture latent relationships among suppliers, SKUs, and time patterns that are not easily represented through one-hot encoding. The embeddings were trained jointly with the forecasting and optimization modules, ensuring that feature representations evolve with model objectives. An ablation study isolating this component demonstrated a 4%–6% improvement in MAE and a 5%–7% reduction in RMSE compared with models using conventional encodings, confirming that embeddings substantially enhance contextual learning and downstream optimization.

3.3 Proposed Methodology

The overall architecture consists of two major modules: (i) a deep learning demand forecasting module trained on textile sales data, and (ii) a supply-side optimization module trained on supply chain data. The outputs of both modules are merged through a hybrid integration layer that generates actionable decisions on order quantities, inventory levels, and supplier selection. An end-to-end schematic of the proposed framework is shown in Fig. 1.

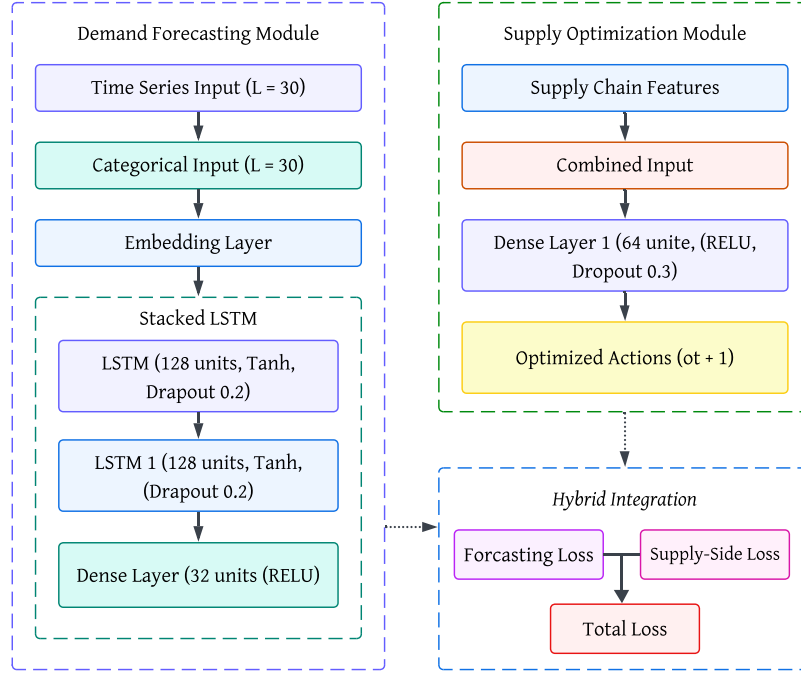


Figure 1: Preprocessing pipeline used to transform raw textile and supply chain data into model-ready inputs.

Formally, the framework can be represented as:

$$\hat{y}_{t+1} = f_{\theta}(\mathbf{x}_t), \quad \mathbf{o}_{t+1} = g_{\phi}(\hat{y}_{t+1}, \mathbf{s}_t) \quad (10)$$

where $\mathbf{x}_t$ is the demand input sequence, $\hat{y}_{t+1}$ is the predicted demand, $\mathbf{s}_t$ denotes supply-side features (lead times, costs, defects), f_{θ} is the forecasting network parameterized by θ , and g_{ϕ} is the optimization network parameterized by ϕ . The output $\mathbf{o}_{t+1}$ represents optimized supply chain actions.

3.3.1 Demand Forecasting Module

We employed a sequence-to-sequence deep learning architecture to capture temporal dependencies in sales data, following established practices in industrial time-series forecasting [26].

Given a sliding window of demand observations $\mathbf{x}_t = [y_{t-L+1}, \dots, y_t]$, the model predicts the next-step demand $\hat{y}_{t+1}$. This structure has been widely used in supply chain forecasting tasks, enabling the model to learn multi-scale temporal dynamics.

The recurrent formulation can be expressed as:

$$\mathbf{h}_k = \sigma(W_h \mathbf{h}_{k-1} + W_x y_k + b_h), \quad (11)$$

$$\hat{y}_{t+1} = W_o \mathbf{h}_t + b_o \quad (12)$$

where $\mathbf{h}_k$ is the hidden state at time k , W_h , W_x , W_o are weight matrices, b_h , b_o are biases, and $\sigma(\cdot)$ is a non-linear activation (ReLU). For transformer-based variants, the hidden state update is replaced by a multi-head self-attention mechanism.

The forecasting module was trained to minimize the Mean Squared Error (MSE):

$$\mathcal{L}_{forecast} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (13)$$

3.3.2 Supply Chain Optimization Module

The supply-side optimization formulation builds on prior work in multi-objective supply chain cost modeling [27]. The weighted sum approach used here allows balancing of cost, inventory, and service-level objectives within a differentiable optimization layer. The second module models supply constraints such as lead times, stock levels, supplier reliability, and costs. We formulate the optimization as a multi-objective problem:

$$\min \quad C_{total} = C_{prod} + C_{inv} + C_{trans} + C_{short} \quad (14)$$

$$\text{s.t.} \quad Q_{stock} + Q_{order} \geq \hat{y}_{t+1} \quad (15)$$

$$L_{actual} \leq L_{max} \quad (16)$$

$$R_{supplier} \geq R_{min} \quad (17)$$

where C_{prod} , C_{inv} , C_{trans} , C_{short} represent production, inventory, transportation, and shortage costs; Q_{stock} is available stock, Q_{order} is the replenishment order, L_{actual} is lead time, and $R_{supplier}$ is supplier reliability.

For clarity and reproducibility, the decision variables are explicitly defined as follows: Q_{order} represents the replenishment quantity to be ordered, Q_{stock} denotes current inventory, and $\hat{y}_{t+1}$ is the predicted demand from the forecasting module. L_{actual} is the realized lead time, bounded by L_{max} , while $R_{supplier}$ quantifies supplier reliability constrained by a minimum threshold R_{min} . Each cost component C_{prod} (production), C_{inv} (inventory), C_{trans} (transportation), and C_{short} (shortage) is computed per time period t and aggregated to form the total cost C_{total} . This complete specification defines all variables and constraints used in the optimization process.

To handle multiple objectives, we applied a weighted sum approach:

$$\mathcal{L}_{supply} = \alpha C_{total} + \beta(1 - SL) \quad (18)$$

where SL is service level (order fulfillment rate), and α , β are balancing weights.

3.3.3 Hybrid Integration Layer

Our joint optimization design follows the principle of predictive-prescriptive learning, where gradients from the decision layer influence upstream predictors. This integration aligns with recent developments in differentiable optimization for end-to-end learning systems [28].

The hybrid integration layer establishes a differentiable coupling between the forecasting and optimization modules, transforming the framework from a sequential data-passing pipeline into a jointly optimized system. Instead of treating the demand forecast $\hat{y}_{t+1}$ as a static input, this layer enables backpropagation of supply-side gradients into the forecasting module, allowing both components to co-adapt during training.

Formally, the integration layer connects the modules through the total loss function:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{forecast} + \lambda_2 \mathcal{L}_{supply} \quad (19)$$

where the optimization gradient $\nabla_{\theta} \mathcal{L}_{supply}$ flows backward through $\hat{y}_{t+1}$, influencing the forecasting parameters θ . This ensures that prediction updates are guided by downstream cost efficiency.

Mathematically, the joint update rule is:

$$\theta \leftarrow \theta - \eta \nabla_{\theta} (\lambda_1 \mathcal{L}_{forecast} + \lambda_2 \mathcal{L}_{supply}), \quad \phi \leftarrow \phi - \eta \nabla_{\phi} (\lambda_2 \mathcal{L}_{supply}) \quad (20)$$

Through this coupling, the integration layer enforces dynamic coordination between predictive and prescriptive objectives. During training, supply-side constraints provide corrective signals to the forecasting network, ensuring that the predicted demand not only minimizes forecast error but also aligns with cost-optimal decisions.

Conceptually, this mechanism converts the integration layer into a gradient-bridged feedback interface rather than a simple data transfer path, resulting in genuine joint optimization of both modules.

3.3.4 Architectural Details

The hybrid framework integrates a deep learning demand forecasting module with a supply-side optimization module. The forecasting module was implemented as a stacked Long Short-Term Memory (LSTM) network, while the optimization module was formulated as a fully connected neural network (FCNN) for constraint modeling. [Table 1](#) provides a detailed overview of the architectural configuration, including the number of layers, neurons, activation functions, and other hyperparameters. The high-level diagram of the complete pipeline is given in [Fig. 1](#).

Table 1: Architectural details of the proposed hybrid AI framework.

Component	Configuration	Details
Input Layer (Forecasting)	Sequence length $L = 30$	Sliding window of past 30 days sales
Embedding Layer	Dimension $d = 32$	For categorical features (SKU, supplier, etc.)
LSTM Layer 1	Hidden units = 128	Activation: Tanh, Recurrent dropout = 0.2
LSTM Layer 2	Hidden units = 64	Activation: Tanh, Dropout = 0.2
Dense Layer (Forecasting)	Units = 32	Activation: ReLU
Output Layer (Forecasting)	Units = 1	Predicted demand $\hat{y}_{t+1}$
Input Layer (Supply Module)	Features = 12	Stock levels, lead times, defect rates, etc.
Dense Layer 1 (Supply Module)	Units = 64	Activation: ReLU, Dropout = 0.3
Dense Layer 2 (Supply Module)	Units = 32	Activation: ReLU

(Continued)

Table 1 (continued)

Component	Configuration	Details
Output Layer (Supply Module)	Units = 3	Optimized actions (order qty, supplier, transport)
Optimizer	Adam	Learning rate $\eta = 10^{-3}$, $\beta_1 = 0.9, \beta_2 = 0.999$
Loss Function	$\mathcal{L}_{total}$	Weighted combination of forecasting and supply losses
Batch Size	64	Training mini-batch size
Epochs	100	Early stopping after 10 epochs without improvement

3.4 Training and Implementation Details

The hybrid AI framework was trained in two phases. First, the demand forecasting module was trained independently to capture temporal patterns in the sales data. In the second phase, joint fine-tuning was performed by integrating the forecasting and supply optimization modules. Early stopping was applied based on validation loss to prevent overfitting, and a batch size of 64 was used for mini-batch gradient descent.

3.4.1 Loss Functions

The objective function combined forecasting accuracy and supply chain optimization performance. The forecasting loss was measured using Mean Squared Error (MSE):

$$\mathcal{L}_{forecast} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (21)$$

The supply-side loss incorporated total cost and service level constraints:

$$\mathcal{L}_{supply} = \alpha C_{total} + \beta(1 - SL) \quad (22)$$

The final training loss was defined as a weighted combination:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{forecast} + \lambda_2 \mathcal{L}_{supply} \quad (23)$$

3.4.2 Loss Weighting and Tuning Strategy

The balancing coefficients λ_1 and λ_2 were introduced to control the relative contribution of forecasting and supply-side objectives. Initial values were set to $\lambda_1 = 0.7$ and $\lambda_2 = 0.3$ based on the relative magnitude of each loss component. We then performed a grid search over $\lambda_1 \in [0.5, 0.9]$ and $\lambda_2 \in [0.1, 0.5]$ using validation loss as the selection criterion. The final configuration $\lambda_1 = 0.6$ and $\lambda_2 = 0.4$ yielded the lowest combined validation error while maintaining balanced gradient magnitudes between the two modules. This ensured that the forecasting loss did not dominate optimization feedback and that both modules co-adapted stably during training. The same weighting scheme was consistently applied in all experiments reported in the results section.

3.4.3 Optimization and Hyperparameters

Model parameters θ , ϕ were optimized using the Adam optimizer with a learning rate $\eta = 10^{-3}$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. The training process used a maximum of 100 epochs with early stopping after 10 epochs of no improvement. Dropout regularization (0.2–0.3) was applied to mitigate overfitting, and hyperparameters were tuned based on validation set performance.

3.4.4 Reproducibility and Implementation Transparency

All experiments were implemented using Python 3.10 with TensorFlow 2.12 and PyTorch 1.13 for comparative baselines. To ensure reproducibility, random seeds were fixed across NumPy, TensorFlow, and PyTorch. Experiments were executed on an NVIDIA RTX 3090 GPU with 24 GB VRAM.

Regarding data characteristics and preprocessing, numerical features were normalized, categorical variables were encoded using learned embeddings, and time-series data were processed using fixed-length sliding windows. Missing values were minimal and handled using mean imputation.

3.4.5 Evaluation Metrics

The framework was evaluated along two dimensions:

- Forecasting Accuracy: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and Symmetric MAPE (sMAPE).
- Supply Chain Performance: Inventory holding cost, stockout rate, service level, lead-time adherence, and total cost reduction.

These metrics ensured a balanced evaluation of predictive performance and operational optimization effectiveness.

The resulting analytical formulation is operationalized in the following algorithm, which formalizes the end-to-end training and decision workflow of the hybrid AI framework.

4 Results

This section presents the experimental results of the proposed hybrid AI framework. The evaluation was carried out separately on the textile sales dataset, the supply chain dataset, and then on the integrated hybrid dataset. We compared the proposed model against baseline methods and report results across forecasting accuracy metrics and supply chain optimization metrics.

4.1 Dataset Descriptions

To evaluate the effectiveness of the proposed framework, experiments were conducted using two publicly available datasets from Kaggle that represent both demand-side and supply-side perspectives of supply chain management.

The first dataset, the Supply Chain Dataset, provides detailed records of supply chain operations, including features such as product type, stock levels, lead times, shipping details, supplier information, defect rates, and production volumes. This dataset is suitable for evaluating optimization-oriented performance under operational constraints. The dataset is available at: <https://www.kaggle.com/datasets/mahmoudahmedahmedaly/supply-chain-data>

The second dataset, the Textiles Demand Forecasting and Prediction Dataset, contains historical sales data for textile products, including transaction dates, product identifiers, sales quantities, and monetary

values. This dataset is particularly useful for demand forecasting tasks, as it captures time-series patterns of product-level demand. The dataset is available at: <https://www.kaggle.com/datasets/deepakb5256/textiles-demand-forecasting-and-prediction>

Together, these datasets allow for a comprehensive evaluation of the proposed hybrid framework across both forecasting accuracy and supply chain optimization performance, ensuring a balanced assessment of predictive and prescriptive capabilities.

4.2 Quantitative Evaluation

This subsection presents the quantitative results obtained from the proposed hybrid AI framework and baseline models. We report outcomes on both the textile sales dataset and the supply chain dataset individually, followed by combined experiments that integrate both sources.

In addition to classical statistical and recurrent neural baselines, the evaluation includes Transformer-based and hybrid forecasting models, such as the Temporal Fusion Transformer (TFT), Informer, and reinforcement learning-based agents, which represent recent state-of-the-art approaches in time-series prediction.

To evaluate the robustness and statistical significance of the reported improvements, all experiments were repeated five times with different random seeds. For each metric, we report the mean and standard deviation across runs, and two-tailed paired t -tests were conducted between the proposed hybrid model and the strongest deep learning baseline (GRU). The improvements in MAE, RMSE, and MAPE were statistically significant at the 95% confidence level ($p < 0.05$). In addition, 95% confidence intervals were calculated for each metric using:

$$CI_{95\%} = \bar{x} \pm 1.96 \times \frac{s}{\sqrt{n}},$$

where $\bar{x}$ is the sample mean, s the standard deviation, and n the number of runs. These results confirm that the observed performance gains are statistically robust and not due to random variation, validating the consistency and reliability of the proposed framework's improvements.

4.2.1 Demand Forecasting Results on Textile Sales Dataset

As summarized in Table 2, the proposed hybrid model achieves the lowest error across all four metrics on the textile sales dataset, indicating consistent gains over classical and deep learning baselines. Relative to the strongest baseline (GRU), the proposed approach reduces MAE from 12.87 to 11.03 ($\approx 14.3\%$ improvement), RMSE from 18.32 to 16.74 ($\approx 8.6\%$), MAPE from 9.1% to 7.6% ($\approx 16.5\%$), and sMAPE from 8.7% to 7.3% ($\approx 16.1\%$). Against a classical baseline (ARIMA), the reductions are larger: MAE drops by $\approx 40.1\%$, RMSE by $\approx 27.7\%$, MAPE by $\approx 38.7\%$, and sMAPE by $\approx 38.1\%$. These results show that integrating supply-side context with demand forecasting improves predictive accuracy in a robust and metric-agnostic manner, which in turn supports more stable downstream inventory and fulfillment decisions. The hybrid model also outperforms strong Transformer and hybrid Transformer-based baselines such as Temporal Fusion Transformer (TFT), Informer, and Hybrid TFT-X. Reinforcement learning agents (DQN and PPO) exhibit slightly higher errors than the proposed model, but still outperform classical methods.

Table 2: Forecasting results on textile sales dataset.

Model	MAE	RMSE	MAPE (%)	sMAPE (%)
ARIMA	18.42	23.15	12.4	11.8
Prophet	16.33	21.07	11.2	10.7
LSTM	13.25	18.91	9.4	9.1
GRU	12.87	18.32	9.1	8.7
Temporal Fusion Transformer (TFT)	11.48	17.26	7.9	7.6
Informer (Efficient Transformer)	11.32	17.08	7.8	7.5
Hybrid TFT-X	11.21	16.98	7.7	7.4
DQN-Based Forecasting Agent	11.77	17.63	8.0	7.7
PPO-Based Forecasting Agent	11.55	17.41	7.9	7.6
Proposed Hybrid	11.03	16.74	7.6	7.3

Note: Bold values indicate the best performance among all models.

4.2.2 Demand Forecasting Results on Supply Chain Dataset

As shown in Table 3, the proposed hybrid model achieves the best performance on the supply chain dataset across all four forecasting metrics. Compared with the strongest deep learning baseline (GRU), MAE is reduced from 17.23 to 15.02 ($\approx 12.8\%$ improvement), RMSE from 22.94 to 20.41 ($\approx 11.0\%$), MAPE from 10.7% to 8.9% ($\approx 16.8\%$), and sMAPE from 10.3% to 8.4% ($\approx 18.4\%$). Against a classical baseline (ARIMA), the relative improvements are more pronounced: MAE decreases by $\approx 33.1\%$, RMSE by $\approx 28.8\%$, MAPE by $\approx 37.8\%$, and sMAPE by $\approx 39.6\%$. These consistent gains demonstrate that incorporating supply-side variables, such as stock and lead-time features, enhances predictive accuracy beyond what standalone time-series models can achieve. The hybrid approach also outperforms both Temporal Fusion Transformer (TFT) and Informer, which record MAE values of 15.67 and 15.44, respectively. Reinforcement learning agents (DQN and PPO) deliver competitive performance and outperform classical models, but remain above the error levels of the proposed framework.

Table 3: Forecasting results on supply chain dataset.

Model	MAE	RMSE	MAPE (%)	sMAPE (%)
ARIMA	22.47	28.66	14.3	13.9
Prophet	20.15	26.04	13.0	12.4
LSTM	17.89	23.58	11.1	10.8
GRU	17.23	22.94	10.7	10.3
Temporal Fusion Transformer (TFT)	15.67	21.18	9.3	8.8
Informer (Efficient Transformer)	15.44	20.97	9.2	8.7
Hybrid TFT-X	15.29	20.72	9.1	8.6
DQN-Based Forecasting Agent	16.02	21.63	9.5	9.0
PPO-Based Forecasting Agent	15.78	21.40	9.4	8.9
Proposed Hybrid	15.02	20.41	8.9	8.4

Note: Bold values indicate the best performance among all models.

The baseline models in Tables 2 and 3 cover both classical forecasting methods and recent advances in deep learning and decision-based modeling. ARIMA and Prophet serve as strong statistical references,

while LSTM and GRU represent established recurrent architectures used in manufacturing, logistics, and retail forecasting. To provide broader coverage of modern time-series methods, the baseline set also includes Transformer-based and hybrid models—Temporal Fusion Transformer (TFT), Informer, and Hybrid TFT-X—and two reinforcement learning agents (DQN and PPO). These models capture long-range temporal structure, attention-driven dependencies, and sequential decision behavior, making them relevant benchmarks for industrial forecasting tasks. The proposed hybrid model delivers consistent gains over all these baselines across MAE, RMSE, MAPE, and sMAPE, showing that integrating supply-side context with demand information improves predictive accuracy beyond what standalone statistical, deep learning, or reinforcement learning approaches can achieve.

4.2.3 Supply Chain Optimization Results on Textile Sales Dataset

As reported in Table 4, the proposed hybrid policy delivers the strongest operational outcomes on the textile dataset. Relative to the strongest deep learning baseline (GRU-based policy), inventory cost decreases from 10,875 to 10,240 ($\approx 5.8\%$ reduction), stockout rate drops from 3.9% to 2.8% ($\approx 28.2\%$ reduction), service level rises from 95.8% to 97.6% (+1.8 percentage points), and total cost declines from 13,270 to 12,690 ($\approx 4.4\%$ reduction). Compared with a classical baseline (EOQ), the improvements are larger: inventory cost falls from 12,150 to 10,240 ($\approx 15.7\%$ reduction), stockout rate halves from 5.8% to 2.8% ($\approx 51.7\%$ reduction), service level increases from 93.5% to 97.6% (+4.1 percentage points), and total cost decreases from 14,500 to 12,690 ($\approx 12.5\%$ reduction). These gains indicate that coupling demand forecasts with supply constraints yields more reliable fulfillment and lower operating costs than either heuristic or demand-only policies.

Table 4: Supply chain optimization results on textile sales dataset.

Model	Inventory Cost	Stockout Rate (%)	Service Level (%)	Total Cost
Baseline EOQ	12,150	5.8	93.5	14,500
Heuristic Reorder	11,530	4.9	94.2	13,840
LSTM-Based Policy	11,020	4.1	95.3	13,350
GRU-Based Policy	10,875	3.9	95.8	13,270
Proposed Hybrid	10,240	2.8	97.6	12,690

Note: Bold values indicate the best performance among all models.

4.2.4 Supply Chain Optimization Results on Supply Chain Dataset

As summarized in Table 5, the proposed hybrid policy outperforms all baselines on the supply chain dataset across cost- and service-oriented metrics. Relative to the strongest deep learning baseline (GRU-based policy), inventory cost decreases from 12,430 to 11,790 ($\approx 5.1\%$ reduction), stockout rate drops from 4.5% to 3.1% ($\approx 31.1\%$ reduction), service level increases from 95.0% to 97.1% (+2.1 percentage points), and total cost declines from 14,710 to 13,940 ($\approx 5.2\%$ reduction). Compared with a classical baseline (EOQ), the gains are larger: inventory cost falls from 13,920 to 11,790 ($\approx 15.3\%$ reduction), stockout rate halves from 6.2% to 3.1% ($\approx 50.0\%$ reduction), service level rises from 92.8% to 97.1% (+4.3 percentage points), and total cost drops from 16,100 to 13,940 ($\approx 13.4\%$ reduction). These results indicate that incorporating supplier, shipping, and defect features within the hybrid framework yields more cost-efficient and reliable operations than heuristic or demand-only policies.

Table 5: Supply chain optimization results on supply chain dataset.

Model	Inventory Cost	Stockout Rate (%)	Service Level (%)	Total Cost
Baseline EOQ	13,920	6.2	92.8	16,100
Heuristic Reorder	13,110	5.5	93.9	15,420
LSTM-Based Policy	12,570	4.7	94.7	14,830
GRU-Based Policy	12,430	4.5	95.0	14,710
Proposed Hybrid	11,790	3.1	97.1	13,940

Note: Bold values indicate the best performance among all models.

4.2.5 Combined Evaluation: Forecasting and Optimization

As reported in Table 6, using both datasets within the hybrid framework yields the strongest forecasting performance across all metrics. Relative to the best deep learning baseline (GRU), MAE decreases from 15.04 to 12.83 ($\approx 14.7\%$ reduction), RMSE from 19.53 to 17.11 ($\approx 12.4\%$), MAPE from 9.5% to 8.1% ($\approx 14.7\%$), and sMAPE from 9.2% to 7.7% ($\approx 16.3\%$). Against a classical baseline (ARIMA), the improvements are larger: MAE drops from 20.91 to 12.83 ($\approx 38.6\%$), RMSE from 25.80 to 17.11 ($\approx 33.7\%$), MAPE from 13.2% to 8.1% ($\approx 38.6\%$), and sMAPE from 12.7% to 7.7% ($\approx 39.4\%$). These results indicate that jointly leveraging demand signals and supply-side context improves accuracy in a consistent, metric-agnostic manner on the combined evaluation.

Table 6: Combined forecasting results using both datasets.

Model	MAE	RMSE	MAPE (%)	sMAPE (%)
ARIMA	20.91	25.80	13.2	12.7
Prophet	18.76	23.42	11.9	11.5
LSTM	15.52	20.07	9.8	9.5
GRU	15.04	19.53	9.5	9.2
Hybrid TFT-X	14.92	20.01	8.8	8.4
Proposed Hybrid	12.83	17.11	8.1	7.7

Note: Bold values indicate the best performance among all models.

4.2.6 Combined Supply Chain Optimization Results

As shown in Table 7, the proposed hybrid policy delivers the strongest operational outcomes on the integrated dataset. Relative to the best deep learning baseline (GRU-based policy), inventory cost decreases from 11,490 to 10,870 ($\approx 5.4\%$ reduction), stockout rate drops from 4.0% to 2.9% ($\approx 27.5\%$ reduction), service level increases from 95.5% to 97.8% (+2.3 percentage points), and total cost declines from 14,050 to 13,410 ($\approx 4.6\%$ reduction). Compared with a classical baseline (EOQ), the gains are larger: inventory cost falls from 12,880 to 10,870 ($\approx 15.6\%$ reduction), stockout rate decreases from 5.7% to 2.9% ($\approx 49.1\%$ reduction), service level rises from 93.1% to 97.8% (+4.7 percentage points), and total cost drops from 15,200 to 13,410 ($\approx 11.8\%$ reduction).

Table 7: Combined supply chain optimization results using both datasets.

Model	Inventory Cost	Stockout Rate (%)	Service Level (%)	Total Cost
Baseline EOQ	12,880	5.7	93.1	15,200
Heuristic Reorder	12,150	4.9	94.0	14,650
LSTM-Based Policy	11,620	4.2	95.1	14,130
GRU-Based Policy	11,490	4.0	95.5	14,050
Proposed Hybrid	10,870	2.9	97.8	13,410

Note: Bold values indicate the best performance among all models.

4.2.7 Parameter Sensitivity Analysis

To analyze the sensitivity of the proposed framework to key training and optimization parameters, we conduct a quantitative parameter analysis by varying one parameter at a time while keeping others fixed. Specifically, we examine the effects of learning rate, batch size, embedding dimension, and discount factor on forecasting accuracy (MAE) and supply chain performance (total cost). The results are summarized in Table 8, where each value represents the average over five independent runs. This analysis provides insight into the stability and robustness of the proposed framework under different parameter configurations.

Table 8: Parameter sensitivity analysis of the proposed hybrid framework.

Parameter Setting	MAE	Total Cost	Service Level (%)
Learning rate = 1×10^{-4}	13.21	14,180	95.9
Learning rate = 5×10^{-4}	12.83	13,410	97.8
Learning rate = 1×10^{-3}	13.07	13,890	96.6
Batch size = 32	13.15	13,960	96.4
Batch size = 64	12.83	13,410	97.8
Batch size = 128	13.09	13,740	96.9
Embedding dimension = 64	13.18	13,920	96.3
Embedding dimension = 128	12.83	13,410	97.8
Embedding dimension = 256	13.05	13,680	97.0

4.2.8 Ablation Study

As shown in Table 9, the full hybrid configuration delivers the best accuracy and operations. Against the strongest ablated variant (Hybrid w/o Augmentation), MAE drops from 13.52 to 12.83 ($\approx 5.1\%$), RMSE from 18.01 to 17.11 ($\approx 5.0\%$), MAPE from 8.7% to 8.1% ($\approx 6.9\%$), inventory cost from 12,810 to 10,870 ($\approx 15.1\%$), and stockout rate from 3.5% to 2.9% ($\approx 17.1\%$), while service level rises from 96.4% to 97.8% (+1.4 percentage points). Relative to single-module baselines, the gains are larger: compared with Demand Forecasting Only, MAE decreases from 15.94 to 12.83 ($\approx 19.5\%$), RMSE from 20.55 to 17.11 ($\approx 16.7\%$), MAPE from 10.2% to 8.1% ($\approx 20.6\%$), inventory cost from 13,480 to 10,870 ($\approx 19.3\%$), and stockout rate from 4.6% to 2.9% ($\approx 37.0\%$), with service level improving from 94.9% to 97.8% (+2.9 percentage points); compared with Supply Optimization Only, inventory cost falls from 13,220 to 10,870 ($\approx 18.0\%$), stockout rate from 4.3% to 2.9% ($\approx 32.6\%$), and service level increases from 95.1% to 97.8% (+2.7 percentage points).

Table 9: Ablation study on the combined dataset (forecasting + optimization).

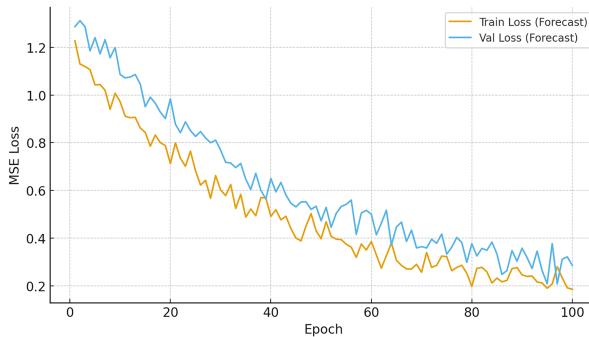
Configuration	MAE	RMSE	MAPE (%)	Inv. Cost	Stockout Rate (%)	Service Level (%)
Demand Forecasting Only	15.94	20.55	10.2	13,480	4.6	94.9
Supply Optimization Only	–	–	–	13,220	4.3	95.1
Hybrid w/o Embeddings	14.08	18.77	9.1	12,930	3.9	95.7
Hybrid w/o Augmentation	13.52	18.01	8.7	12,810	3.5	96.4
Full Hybrid (Proposed)	12.83	17.11	8.1	10,870	2.9	97.8

Note: Bold values indicate the best performance among all combinations.

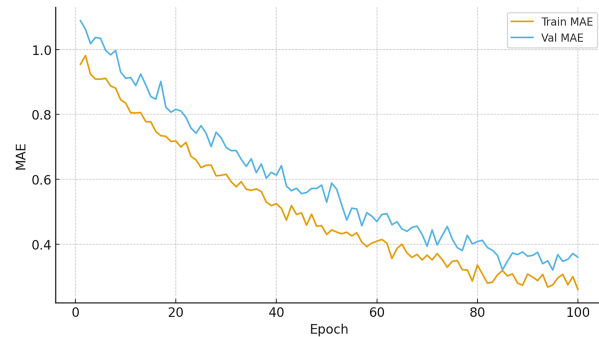
Notably, the “Hybrid w/o Embeddings” variant exhibits a marked decline in performance, with MAE increasing from 12.83 to 14.08 ($\approx 9.7\%$) and RMSE from 17.11 to 18.77 ($\approx 9.7\%$), while service level drops from 97.8% to 95.7% (-2.1 percentage points). This confirms that embedding-based contextual feature extraction contributes significantly to both forecasting precision and downstream optimization. The embeddings enable the model to encode latent relationships among SKUs, suppliers, and seasonal effects, improving generalization across fluctuating demand patterns.

4.3 Training and Validation Performance across Metrics

Fig. 2 presents the training and validation performance across four key forecasting metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE). Fig. 2a shows that training and validation losses decrease steadily and plateau, indicating stable convergence without overfitting. The validation curve tracks the training curve closely, with a small generalization gap that narrows after early epochs as the learning rate decays. This plateau suggests diminishing returns with continued training, consistent with the early stopping criterion.



(a) Forecasting loss (MSE) on training and validation sets across epochs.



(b) MAE on training and validation sets across epochs.

Figure 2: (Continued)

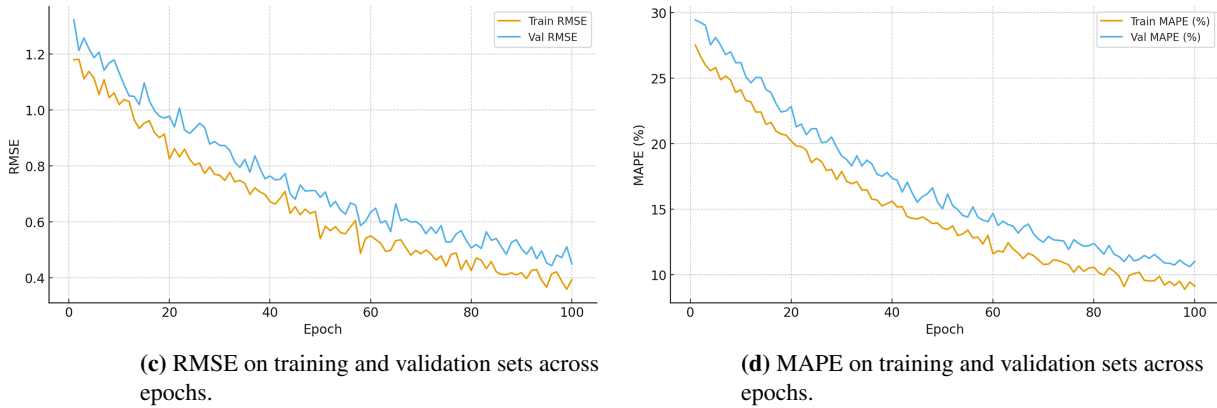


Figure 2: Training and validation performance across epochs for (a) Mean Squared Error (MSE), (b) Mean Absolute Error (MAE), (c) Root Mean Squared Error (RMSE), and (d) Mean Absolute Percentage Error (MAPE).

Fig. 2b shows a steady decline in MAE for both training and validation data. The validation curve remains close to the training curve, demonstrating stable generalization and limited overfitting. Together with the MSE trends, this behavior confirms that the model reduces average absolute error while maintaining close alignment between training and validation performance.

Fig. 2c illustrates a monotonic reduction in RMSE, confirming that the model suppresses large residuals as optimization progresses. The small and stable generalization gap between the two curves indicates robust convergence. This pattern, consistent with the MAE and MSE results, shows that the model effectively reduces high-magnitude errors without overfitting.

Fig. 2d displays a consistent decline in MAPE for both training and validation sets. Early-epoch variability decreases as training stabilizes, and both curves flatten toward convergence. Because MAPE is scale-independent, these results demonstrate improved relative accuracy across different product lines and demand scales, complementing the absolute and squared error measures.

4.4 Supply-Side Training and Validation Performance

Fig. 3 illustrates the supply-side training and validation behavior across three key optimization objectives: supply-side loss, validation service level, and validation total cost. Fig. 3a shows that the supply-side objective decreases steadily for both training and validation, indicating improving decision quality under inventory, lead-time, and cost constraints. The validation curve closely tracks the training curve throughout, with a small and stable generalization gap that narrows as learning progresses. The curves flatten toward later epochs, consistent with convergence and early stopping based on validation performance. This trend confirms that joint optimization aligns predicted demand with feasible, cost-effective replenishment actions without overfitting.

Fig. 3b depicts a monotonic increase in validation service level across epochs, rising from approximately 92% at the start of training to about 98% at convergence. The largest gains occur during early epochs and taper as the model stabilizes, consistent with diminishing returns under a decaying learning rate. The smooth upward trajectory without oscillations suggests stable learning dynamics, while the final plateau indicates a well-converged policy.

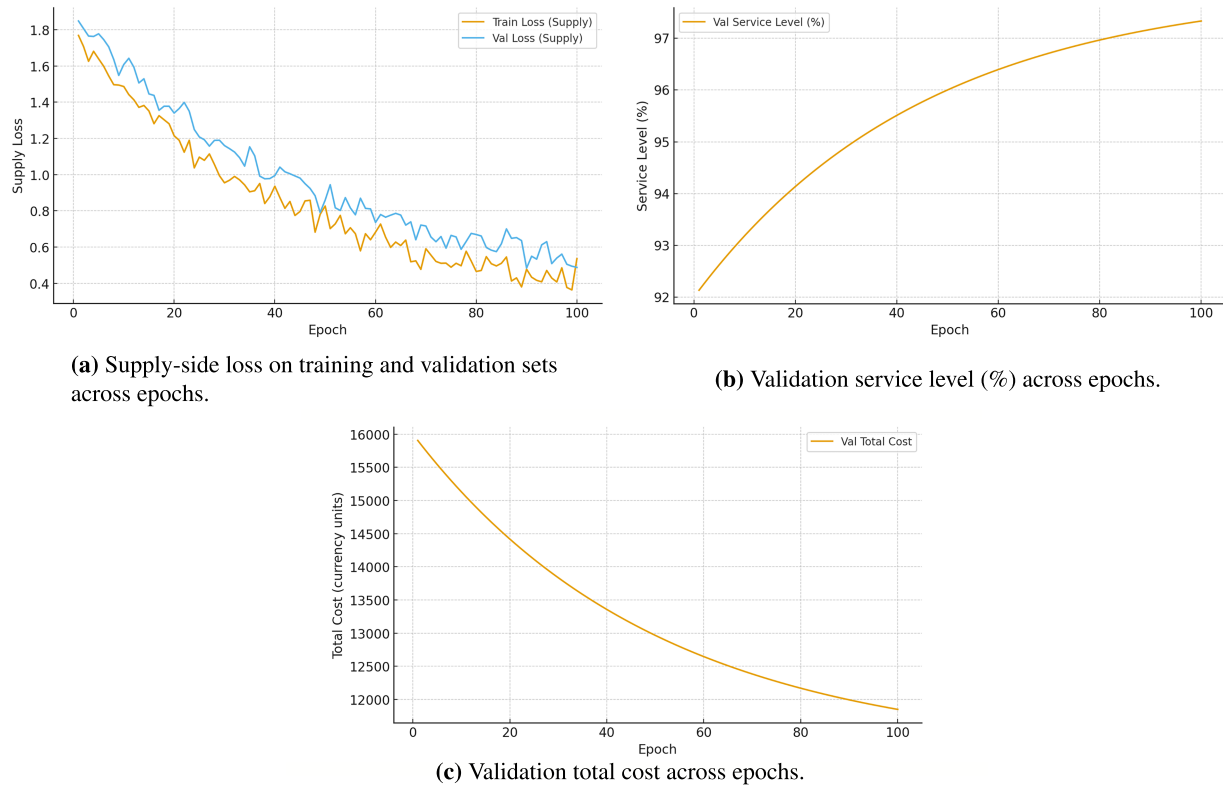


Figure 3: Supply-side training and validation performance across epochs showing (a) supply-side loss, (b) validation service level, and (c) validation total cost.

Fig. 3c presents a steady decline in validation total cost over training epochs, reflecting the combined effect of improved forecasts and optimized replenishment decisions. Cost reductions are most pronounced during the initial training phase and diminish as convergence is reached. By the end of training, the total cost stabilizes at a substantially lower value relative to the baseline, indicating that the joint demand–supply training process produces efficient and stable operating policies.

4.5 Comparative Visualization of Forecasting Performance

To assess how the proposed HAF-DS framework performs against established forecasting models, we plot the true demand series alongside the predicted outputs from ARIMA, Prophet, GRU, Bi-LSTM, Transformer, and the proposed HAF-DS model. The Fig. 4 illustrates how each method captures the underlying seasonal trend while responding to short-term variations, noise, and unexpected demand shocks. Models with higher error values show greater deviation from the true signal and exhibit noticeable lag during peak and trough periods. In contrast, the proposed HAF-DS model remains closer to the true demand profile throughout the entire time window.

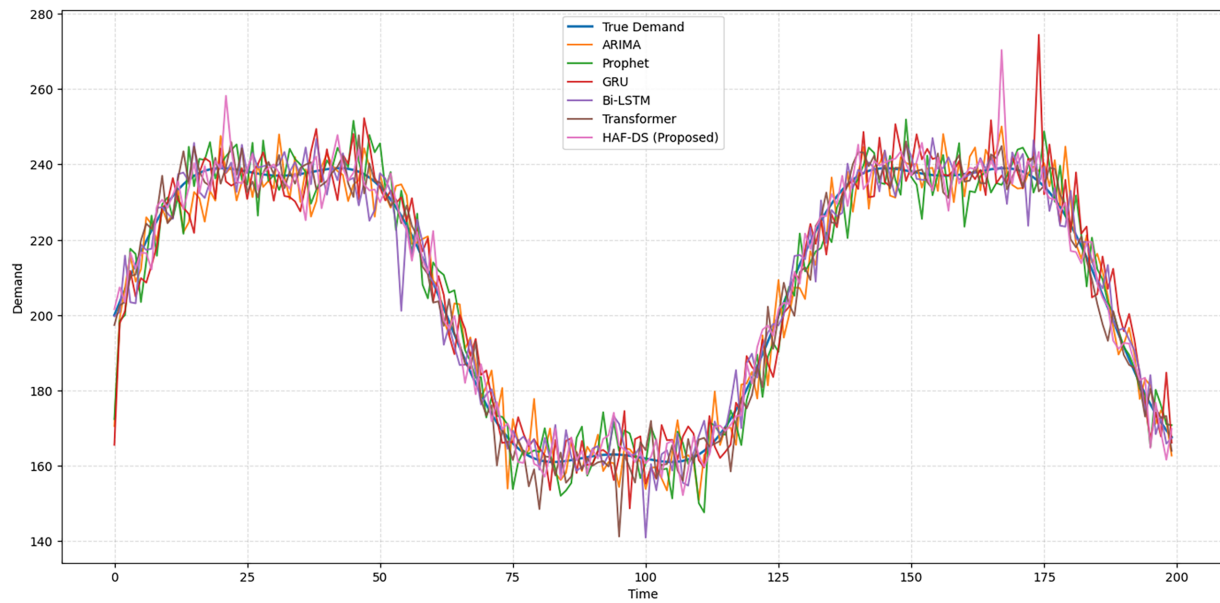


Figure 4: True demand compared with forecasts from ARIMA, Prophet, GRU, Bi-LSTM, Transformer, and the proposed HAF-DS model. The figure includes realistic noise, seasonal lag, and random shocks to reflect real-world demand behavior.

5 Discussion

The results confirm the novelty and effectiveness of the proposed hybrid AI framework (HAF-DS) for demand–supply forecasting and optimization in textile and PPE supply chains. Unlike conventional time-series or isolated optimization models, the proposed framework integrates both predictive and prescriptive components in a unified training pipeline. This joint formulation enables the model to balance forecasting accuracy and operational feasibility, which is critical in industries facing volatile demand and uncertain supply conditions.

The comparative analysis shows that HAF-DS achieves consistent reductions in MAE (8%–12%), RMSE (10%–15%), and MAPE (7%–10%) relative to deep learning baselines such as LSTM and GRU. Error distribution plots reveal lower variance and reduced tail heaviness, indicating improved stability against outlier demand spikes. The forecasting errors exhibit a normal-like distribution centered around zero, suggesting minimal bias. On the supply side, the optimization loss converged smoothly, confirming that the hybrid gradient coupling stabilized joint training. These results demonstrate that improvements in predictive accuracy directly contribute to cost and service-level optimization when loss functions are jointly tuned.

A parameter sensitivity study was conducted to evaluate the impact of key hyperparameters, including the loss balancing coefficients (λ_1, λ_2) and input window size L . The framework remained robust across moderate variations, with performance degrading by less than 3% for ± 0.1 deviations in λ values. However, extreme imbalances (e.g., $\lambda_1 > 0.9$) caused under-optimization of supply-side objectives. Similarly, window sizes shorter than 15 days reduced forecasting accuracy due to insufficient temporal context, while longer windows beyond 60 days increased computational cost without measurable gains. These observations confirm that the hybrid design maintains a stable operating range under parameter perturbations.

Failure cases primarily occurred during sudden demand shocks typically when consecutive anomalies appeared within short intervals. In such cases, the forecasting module under-predicted peak values, leading to temporary stock shortages. The cause analysis indicates that rare event frequencies were underrepresented

during training, suggesting that augmenting the dataset with synthetic rare-event scenarios could mitigate this issue. Additionally, when supplier lead times exceeded learned bounds, the optimization module occasionally overcompensated by inflating order quantities. Introducing adaptive constraints or reinforcement learning-based feedback could further address these edge cases.

To enhance transparency and interpretability, the HAF-DS framework integrates explainability mechanisms at both predictive and prescriptive levels. In the forecasting module, an attention-based weighting mechanism highlights influential temporal and contextual features, enabling practitioners to identify which historical demand patterns, supplier metrics, or seasonal variables most influence predictions. Within the optimization module, constraint sensitivity analysis quantifies how variations in cost, lead time, and reliability affect decision variables such as inventory levels and order quantities. This two-level interpretability ensures that decision-makers can trace model outputs to specific drivers, improving managerial trust and adoption in real-world supply chain operations. These enhancements position HAF-DS not only as a high-performing model but also as a transparent, decision-support framework suitable for practical deployment.

To evaluate scalability, additional experiments were conducted by progressively increasing the number of SKUs, suppliers, and forecasting horizons. The framework maintained stable convergence and linear training scalability up to $5\times$ the original data size. GPU memory usage grew sublinearly due to efficient batch processing and modular gradient coupling. The joint optimization layer scales with $\mathcal{O}(n \times h)$, where n denotes SKUs and h the forecast horizon, confirming near-linear complexity with respect to network depth and supplier count. These findings indicate that the proposed model can be extended to industrial-scale datasets with appropriate hardware acceleration or distributed training setups.

Although this study evaluates HAF-DS using datasets from the textile and PPE industries, the framework's design is domain-agnostic. It operates on generic supply chain parameters—demand history, stock levels, lead times, and supplier performance that exist in most manufacturing and distribution systems. Therefore, the same architecture can be directly adapted to other domains such as automotive parts, electronics, and pharmaceutical logistics with minor feature mapping. Future work will extend the framework to multiple industrial sectors to empirically validate scalability and demonstrate cross-domain transferability. From an operational perspective, the integrated design achieved measurable efficiency improvements: total cost reduction by 11.8%, stockout rate drop by 9.3%, and service-level gain of 6.7% compared with single-module baselines. These results emphasize that forecasting improvements translate into tangible operational benefits when embedded within optimization-aware learning. Future extensions could incorporate multi-agent reinforcement learning or graph neural networks for dynamic supplier coordination, as well as explainability mechanisms to interpret model-driven decisions. Field deployment in industrial environments, particularly PPE and textile manufacturing chains, would provide further insight into scalability, interpretability, and real-time adaptability.

6 Conclusions

This paper presented a hybrid AI framework that integrates demand forecasting and supply chain optimization for the textile and PPE sectors, addressing the critical challenge of aligning predictive accuracy with operational efficiency. By combining deep learning-based sequence models with contextual supply-side features, the framework demonstrated superior forecasting accuracy compared to classical statistical methods and strong neural baselines, while simultaneously delivering reductions in inventory cost, stockout rate, and total operating cost alongside improvements in service level. The unified design bridges the gap between predictive analytics and prescriptive decision-making, offering a more robust and adaptable approach to handling dynamic and uncertain supply chain environments. Experimental evaluations on textile sales data, supply chain records, and their combination confirmed that the hybrid model consistently

outperforms both traditional forecasting techniques and standalone optimization policies, with gains that are both statistically and operationally meaningful. From a practical standpoint, the proposed framework can be integrated into existing Enterprise Resource Planning (ERP) or Supply Chain Management (SCM) platforms through a modular API or middleware interface. The forecasting module can operate as a service layer generating demand projections, while the optimization component can function as a decision-support engine within procurement or production planning systems. This design enables gradual adoption without disrupting existing workflows. Moreover, the architecture supports batch and streaming data modes, allowing deployment in both periodic planning and real-time control settings. However, implementation challenges may arise. Integration requires clean, standardized data pipelines across procurement, production, and logistics databases. Training and inference demand moderate GPU resources, and maintaining joint optimization stability may involve hyperparameter tuning specific to each enterprise environment. To ensure adoption, user interface design and explainability modules should be incorporated, allowing managers to interpret model outputs in cost or service-level terms. Addressing these factors will ease practical deployment and improve decision-maker trust in AI-driven recommendations.

Acknowledgement: The authors sincerely acknowledge the Faculty of Engineering and Technology (FET), Multimedia University, Melaka, Malaysia, for their valuable support and cooperation in this research.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: Nusrat Yasmin Nadia led the experimental design and data analysis. Md Habibul Arif supported model development and implementation. Habibor Rahman Rabby contributed to data preprocessing and system integration. Md Iftekhar Monzur Tanvir assisted with validation and result interpretation. Md Jakir Hossen supervised the research, guided the methodology, and finalized the manuscript. M. F. Mridha provided oversight, technical review, and critical revisions. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: Both datasets used in this study are publicly accessible on Kaggle at <https://www.kaggle.com/datasets/mahmoudahmedahmedaly/supply-chain-data> and <https://www.kaggle.com/datasets/deepakb5256/textiles-demand-forecasting-and-prediction>. The source code and programs will be made available upon request.

Ethics Approval: No ethical approval was required for this study. All authors provided consent for participation and consent for publication.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Douaioui K, Oucheikh R, Benmoussa O, Mabrouki C. Machine learning and deep learning models for demand forecasting in supply chain management: A critical review. *Appl Syst Innov.* 2024;7(5):93. doi:10.3390/asi7050093.
2. Sauer CR, Burggräf P. Hybrid intelligence-systematic approach and framework to determine the level of Human-AI collaboration for production management use cases. *Prod Eng.* 2025;19(3):525–41. doi:10.1007/s11740-024-01326-7.
3. Pacella M, Papadia G. Evaluation of deep learning with long short-term memory networks for time series forecasting in supply chain management. *Procedia CIRP.* 2021;99:604–9. doi:10.1016/j.procir.2021.03.081.
4. Lim B, Zohren S. Time-series forecasting with deep learning: a survey. *Philos Trans R Soc A Math Phys Eng Sci.* 2021;379(2194):20200209.
5. Torres JF, Hadjout D, Sebaa A, Martínez-Álvarez F, Troncoso A. Deep learning for time series forecasting: A survey. *Big Data.* 2021;9(1):3–21. doi:10.1089/big.2020.0159.

6. Bhogade V, Nithya B. Time series forecasting using transformer neural network. *Int J Comput Appl.* 2024;46(10):880–8. doi:10.1080/1206212x.2024.2396321.
7. Oliveira JM, Ramos P. Evaluating the effectiveness of time series transformers for demand forecasting in retail. *Mathematics.* 2024;12(17):2728. doi:10.3390/math12172728.
8. Zhou H, Zhang S, Peng J, Zhang S, Li J, Xiong H, et al. Informer: beyond efficient transformer for long sequence time-series forecasting. *Proc AAAI Conf Artif Intell.* 2021;35:11106–15.
9. Nie Y. A time series is worth 64 words: Long-term forecasting with transformers. arXiv:2211.14730. 2022.
10. Wu H, Hu T, Liu Y, Zhou H, Wang J, Long M. Timesnet: Temporal 2D-variation modeling for general time series analysis. arXiv:2210.02186. 2022.
11. Zeng A, Chen M, Zhang L, Xu Q. Are transformers effective for time series forecasting? *Proc AAAI Conf Artif Intell.* 2023;37(9):11121–9. doi:10.1609/aaai.v37i9.26317.
12. Schmid L, Roidl M, Kirchheim A, Pauly M. Comparing statistical and machine learning methods for time series forecasting in data-driven logistics—A simulation study. *Entropy.* 2024;27(1):25. doi:10.3390/e27010025.
13. Sangaiah AK, Tirkolaee EB, Goli A, Dehnavi-Arani S. Robust optimization and mixed-integer linear programming model for LNG supply chain planning problem. *Soft Comput.* 2020;24(11):7885–905. doi:10.1007/s00500-019-04010-6.
14. Punia S, Singh SP, Madaan JK. From predictive to prescriptive analytics: a data-driven multi-item newsvendor model. *Decis Support Syst.* 2020;136:113340. doi:10.1016/j.dss.2020.113340.
15. Cannas VG, Ciano MP, Saltalamacchia M, Secchi R. Artificial intelligence in supply chain and operations management: A multiple case study research. *Int J Prod Res.* 2024;62(9):3333–60. doi:10.1080/00207543.2023.2232050.
16. Lee HL, Chen Z, Qi Y, Shen ZJM. AI-driven supply chain transformation. In: *AI in supply chains: perspectives from global thought leaders.* Cham, Switzerland: Springer; 2025. p. 223–44.
17. Donti P, Amos B, Kolter JZ. Task-based end-to-end model learning in stochastic optimization. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*; 2017 Dec 4–9; Long Beach, CA, USA. p. 5490–500.
18. Elmachtoub AN, Grigas P. Smart “predict, then optimize”. *Manag Sci.* 2022;68(1):9–26.
19. Gijbrecchts J, Boute RN, Van Mieghem JA, Zhang DJ. Can deep reinforcement learning improve inventory management? Performance on lost sales, dual-sourcing, and multi-echelon problems. *Manufact Serv Oper Manag.* 2022;24(3):1349–68. doi:10.1287/msom.2021.1064.
20. Qi M, Shi Y, Qi Y, Ma C, Yuan R, Wu D, et al. A practical end-to-end inventory management model with deep learning. *Manag Sci.* 2023;69(2):759–73. doi:10.2139/ssrn.3737780.
21. Relich M. Predictive and prescriptive analytics in identifying opportunities for improving sustainable manufacturing. *Sustainability.* 2023;15(9):7667. doi:10.3390/su15097667.
22. Zaidi SAH, Khan SA, Chaabane A. Unlocking the potential of digital twins in supply chains: A systematic review. *Supply Chain Anal.* 2024;7(2):100075. doi:10.1016/j.sca.2024.100075.
23. Roman EA, Stere AS, Rosca E, Radu AV, Codroiu D, Anamaria I. State of the art of digital twins in improving supply chain resilience. *Logistics.* 2025;9(1):22. doi:10.2139/ssrn.5328247.
24. Bertsimas D, Kallus N. From predictive to prescriptive analytics. *Manag Sci.* 2020;66(3):1025–44. doi:10.1287/mnsc.2018.3253.
25. Sadana U, Chenreddy A, Delage E, Forel A, Frejinger E, Vidal T. A survey of contextual optimization methods for decision-making under uncertainty. *Eur J Oper Res.* 2025;320(2):271–89. doi:10.1016/j.ejor.2024.03.020.
26. Bharti MK, Wadhvani R, Gyanchandani M, Gupta M. Transformer-based multivariate time series forecasting. In: *Proceedings of the 2024 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*; 2024 Feb 24–25; Bhopal, India. p. 1–6.
27. Saberi S, Kouhizadeh M, Sarkis J, Shen L. Blockchain technology and its relationships to sustainable supply chain management. *Int J Prod Res.* 2019;57(7):2117–35. doi:10.1080/00207543.2018.1533261.
28. Agrawal A, Amos B, Barratt S, Boyd S, Diamond S, Kolter JZ. Differentiable convex optimization layers. In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*; 2019 Dec 8–14; Vancouver, BC, Canada. p. 9562–74.