Journal of Computer Technology and Applied Mathematics
Journal Home: http://jctam.suaspress.org | CODEN: JCTAAD
Vol. 2, No. 6, 2025 | ISSN 3007-4126 (Print) | ISSN 3007-4134 (Online) | ISSN 3007-4142 (Digital)

LSTM-Based Deep Learning Models for Long-Term Inventory Forecasting in Retail Operations

HUANG, Sichong [1*]

[1] Duke University, USA

*** HUANG, Sichong is the corresponding author, E-mail: sichong.huang@alumni.duke.edu***

Abstract: Given the complex fluctuations and extended forecasting cycles inherent in retail inventory, this study investigates the application of LSTM deep learning models for inventory time series forecasting. It details the model architecture design, parameter optimization, and training methodology, while presenting the data construction and experimental validation process. Results demonstrate that the model effectively captures inventory variation patterns, enhances prediction accuracy and trend stability, and exhibits strong generalization capabilities across multiple forecasting horizons.

Keywords: Retail Inventory, LSTM Model, Time Series Forecasting, Parameter Optimization.

Disciplines: Artificial Intelligence. **Subjects:** Deep Learning.

DOI: https://doi.org/10.70393/6a6374616d.333238 **ARK:** https://n2t.net/ark:/40704/JCTAM.v2n6a04

1 INTRODUCTION

Retail inventory is significantly influenced by seasonal patterns, promotions, and supply chain fluctuations, making it challenging for traditional statistical models to capture long-term temporal characteristics. To address nonlinear correlations and trend lags in inventory forecasting, an LSTM-based deep learning model was developed to enable dynamic learning of multidimensional features and identification of periodic patterns. This study encompasses data construction, model optimization, and error analysis, aiming to enhance prediction accuracy and stability, thereby providing technical support for inventory control and supply chain decision-making in retail enterprises.

2 INVENTORY FORECASTING MODELING

Retail inventory levels exhibit pronounced nonlinear fluctuations driven by seasonal sales patterns, price adjustments, and promotional activities, rendering traditional ARIMA models inadequate for capturing multi-cycle characteristics in long-term time series (Wang & Wang, 2025) [1]. To achieve precise forecasting of inventory trends over 30–90 days, a time-series sliding-window-based inventory prediction model is established. With a time step of n, the input sequence incorporates multidimensional features including historical sales volume, procurement volume, promotion intensity, and holiday dummy variables. These are transformed into prediction outputs via the time-series mapping function $f(\cdot)$. The model objective minimizes the deviation between the predicted inventory $\hat{y}_{t+k}$ and the actual inventory y_{t+k}, defined as:

$$\hat{y}_{t+k} = f\left(x_t, x_{t-1}, \ldots, x_{t-n+1}; \theta\right) \quad (1)$$

Where: θ represents the model parameter set, and x_t denotes the feature vector at time t.

Data samples were selected from the sales records of a large retail chain from 2021 to 2024, with daily SKU inventory fluctuations ranging from 1,200 to 2,400 units. This modeling framework provides input structure and objective constraints for subsequent LSTM model design, enabling the transition from statistical forecasting to deep temporal learning.

3 LSTM MODEL DESIGN

3.1 Network Architecture

Addressing the strong temporal dependency and pronounced seasonal fluctuations characteristic of retail inventory forecasting (Rong & Vinay, 2024) [2], the LSTM architecture adopts a "multi-layer gated recurrent-fully connected output" structure (see Figure 1). The input layer receives multidimensional feature sequences spanning the past 60 days. Each sample comprises eight dimensions including sales volume, procurement volume, holiday indicators, and price fluctuations, which are embedded into an input matrix $X \in R^{60\times 8}$. The backbone consists of two stacked LSTM layers with 128 and 64 hidden units

Journal of Computer Technology and Applied Mathematics
Journal Home: http://jctam.suaspress.org | CODEN: JCTAAD
Vol. 2, No. 6, 2025 | ISSN 3007-4126 (Print) | ISSN 3007-4134 (Online) | ISSN 3007-4142 (Digital)

respectively, enabling sequential extraction of long-term and medium-term inventory dynamics. To prevent gradient vanishing and enhance generalization, a 0.2 Dropout mechanism and residual connections are introduced between layers (Ming et al., 2023) [3].

At the unit level, state transitions follow:

$$h_t = o_t * \tanh\left(f_t * C_{t-1} + i_t * \widetilde{C}_t\right) \quad (2)$$

where: f_t, i_t, and o_t represent the forget, input, and output gate control factors, respectively, and C_t denotes the memory cell state.

The output layer employs a linear activation function to achieve continuous forecasting of future inventory levels $\hat{y}_{t+k}$. The network comprises approximately 8.6×10^5 parameters, constructed using the TensorFlow framework with GPU parallel training support. This architecture balances long-term trend memory with short-term volatility response, providing a stable feature extraction foundation for subsequent parameter optimization (Yang & Yu, 2023) [4].

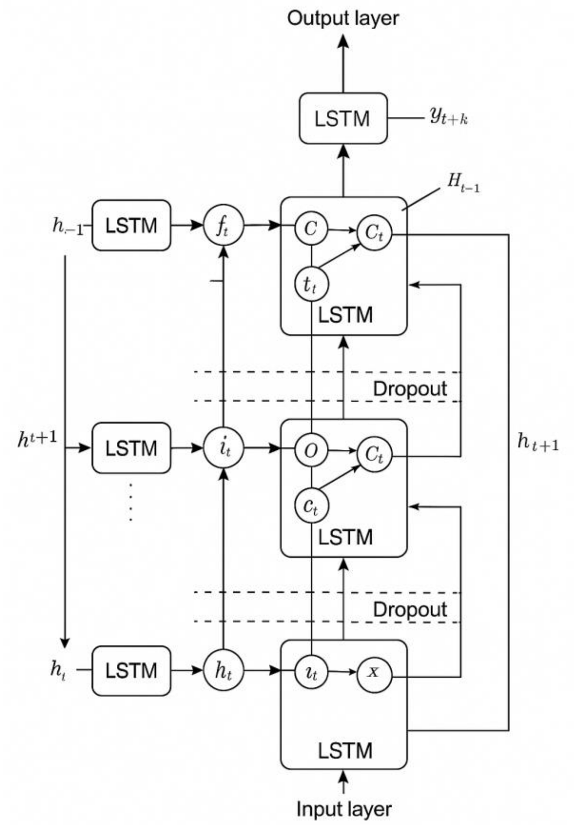

FIGURE 1. SCHEMATIC DIAGRAM OF LSTM NETWORK ARCHITECTURE

3.2 PARAMETER OPTIMIZATION

To improve the stability and generalization of the LSTM model in long-term inventory forecasting, key hyperparameters were systematically designed and optimized. The process followed a three-phase approach—"theoretical constraints, grid search, and validation evaluation"—to balance model complexity and temporal feature extraction capability. Based on a two-layer LSTM structure, preliminary tuning was conducted on hidden units, time steps, and Dropout ratios. When the time window was set to 60 days and hidden units to 128 and 64, the model achieved stable performance in capturing sales and replenishment cycles, with about 8.6×10^5 parameters ensuring a balance between accuracy and computational efficiency.

Adam was adopted as the optimizer with an initial learning rate of 0.001 and a decay factor of 0.95. Batch size ranged between 32 and 64, and the MSE loss function was used with Early Stopping to avoid overfitting. Dropout rates of 0.2 – 0.3 and L2 regularization further enhanced generalization by reducing gradient oscillation. All hyperparameters were finalized through three-fold cross-validation in TensorFlow 2.15. Table 1 summarizes the main hyperparameters and their optimal values.

TABLE 1. LSTM MODEL KEY HYPERPARAMETERS

Parameter Name	Description	Value/Range	Unit/Type
Time Steps	Input Sequence Length	60	Days
Hidden Units (Layer 1)	Number of neurons in first LSTM layer	128	Units
Hidden Units (Layer 2)	Number of neurons in second LSTM layer	64	Units
Dropout Rate	Random deactivation ratio	0.2–0.3	–
Learning Rate	Initial learning rate for Adam optimizer	0.001 (decay 0.95)	–
Batch Size	Training batch size	32–64	Samples
Regularization	L2 weight decay coefficient	0.0005	–
Epochs (Max)	Maximum training iterations	200	–
Early Stopping Patience	Epochs without improvement before stopping	10	–
Optimizer	Optimization algorithm used	Adam	–

Journal of Computer Technology and Applied Mathematics
Journal Home: http://jctam.suaspress.org | CODEN: JCTAAD
Vol. 2, No. 6, 2025 | ISSN 3007-4126 (Print) | ISSN 3007-4134 (Online) | ISSN 3007-4142 (Digital)

3.3 Model Training

After finalizing the network architecture and hyperparameters, the model training phase aims to establish an optimal mapping between time-series inventory data and forecast outputs. The process follows a four-step strategy—"data stratification, model initialization, batch iteration, and validation monitoring" — to ensure forecast stability and generalization. The 2021–2024 retail sales and inventory data are chronologically divided into 70% training, 20% validation, and 10% testing sets to maintain temporal continuity. Input sequences undergo normalization to form the tensor sample matrix ` $X \in R^{N\times60\times8}$ `, where ` $N$ `. The total number of samples N is approximately 1.5×10^4, representing a 60-day feature window.

Training utilizes the Adam optimizer with an initial learning rate of 0.001, decaying exponentially by 0.95 every 20 epochs. Batch size is 64, with a maximum of 200 iterations. Overfitting is controlled by real-time validation monitoring and Early Stopping, which halts training if validation loss stagnates for 10 consecutive epochs. The mean squared error (MSE) loss function is employed to measure the deviation between the predicted inventory level $\hat{y}_i$ and the actual inventory level y_i , defined as:

$$L_{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{y}_i - y_i\right)^2 \quad (3)$$

This loss guides parameter updates through backpropagation to minimize overall prediction error (Myungsoo et al., 2022) [5].

A dynamic logging mechanism records training and validation loss curves and model checkpoints, automatically saving the optimal model upon minimum validation error. GPU acceleration on an NVIDIA RTX 3090 (24 GB VRAM) achieves an average iteration time of about 35 seconds. To maintain consistency, random seeds are fixed and batch order preserved. The entire process runs on TensorFlow 2.15, with 3-fold cross-validation used to evaluate generalization across different product categories.

4 DATA PREPARATION

4.1 Dataset Construction

The dataset was built using real operational data from a retail enterprise, covering 42 consecutive months of sales and inventory records from January 2021 to June 2024. It contains about 450,000 entries, including daily sales, replenishment quantities, promotional intensity, product prices, holiday indicators, and weather indices. Data were sourced from the enterprise's ERP and POS systems with daily granularity, accurately capturing inventory dynamics and consumption cycles.

Products were categorized into five groups—food, daily chemicals, apparel, home appliances, and cultural/sports goods—using SKU as the minimum unit, and representative samples were drawn from each category. Sliding windows were then applied chronologically, where each 60-day sequence served as input to forecast inventory levels for the next 30 days, resulting in roughly 1.5×10^4 non-overlapping training samples to preserve temporal continuity.

To ensure adaptability to real market variations, the dataset retained periods with promotions and price adjustments, improving diversity and realism. The finalized dataset was stored in CSV format with structured feature labels and timestamp indices, forming a standardized, high-quality foundation for subsequent preprocessing and model training.

4.2 Data Preprocessing

Following dataset construction, the preprocessing phase focused on removing noise and anomalies to ensure consistency and computational validity of input features. Completeness checks and time alignment were first applied to ERP-exported sales and inventory data. Missing values ($\approx$ 2.3%) were imputed using a hybrid method combining temporal interpolation and weighted averaging across similar SKUs to maintain sequence continuity. Abnormal inventory spikes beyond the $\pm 3\sigma$ range were corrected through median smoothing to reduce distortion from short-term fluctuations.

Categorical attributes such as holidays, promotions, and weather conditions were transformed using one-hot encoding, expanding to 11 discrete dimensions to help the model identify market-driven temporal effects. Continuous variables, including sales, price, and inventory, were normalized to the [0,1] range to standardize scale differences. The normalization is defined as:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (4)$$

Where: ` $x_{\min}$ ` and ` $x_{\max}$ ` represent the minimum and maximum values of the feature, respectively.

The processed features were then organized into tensor format in chronological order, with dimensions (samples, time step = 60, feature dimension = 18), and paired with corresponding prediction labels. Data were stored following a 7:2:1 split for training, validation, and testing, with consistent file indexing and standardized structure to ensure smooth model integration and reliable verification.

5 EXPERIMENTS AND ANALYSIS

5.1 Experimental Setup

To validate the long-term inventory prediction stability and applicability of the LSTM model, the experimental project follows reproducibility and multi-dimensional

Journal of Computer Technology and Applied Mathematics
Journal Home: http://jctam.suaspress.org | CODEN: JCTAAD
Vol. 2, No. 6, 2025 | ISSN 3007-4126 (Print) | ISSN 3007-4134 (Online) | ISSN 3007-4142 (Digital)

verification principles. The experimental platform operates on Windows 11 using Python 3.10 and the TensorFlow 2.15 framework. Hardware specifications include an Intel i9-13900K processor, 64GB RAM, and an NVIDIA RTX 3090 GPU (24GB VRAM), with a single training cycle taking approximately 35 seconds.

The dataset utilized retail sales samples from 2021 to 2024. After normalization and splitting via a moving window, approximately 1.5×10^4 training samples were generated. These samples were sequentially split into a training set (70%), validation set (20%), and test set (10%). The batch size was set to 64, and the maximum training iterations were 200. The learning rate was initialized at 0.001 and decreased exponentially.

To minimize random bias due to sampling dependency, a fixed random seed value and batch resampling were employed during training. Model performance was evaluated using three metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), measuring overall fit, sample variability, and relative error, respectively.

During validation, error metrics are monitored in real-time. When validation loss reaches its optimal point, model weights are automatically saved for loading during the testing phase. All files and experiment records are managed through a unified directory system, ensuring traceability and enabling comparative analysis. This provides a reliable foundation for subsequent model performance comparisons and optimization.

5.2 Results Analysis

To validate the accuracy and stability of the LSTM model in long-term inventory forecasting, quantitative analysis was conducted on experimental results across different forecast horizons (30 days, 60 days, 90 days). Table 2 lists key error metrics on the test set. Results indicate the model performs best in short-term forecasting.The results show that the model performs excellently in short-term forecasts, with errors gradually increasing as the forecast period increases, but remaining, in general, within acceptable limits.

Table 2. LSTM Model Performance on Different Forecast Horizons

Forecast Horizon (Days)	RMSE (Units)	MAE (Units)	MAPE (%)
30	142.6	93.4	5.8
60	187.2	121.9	7.4
90	231.5	162.7	9.1

Table 2 shows that the 30-day forecast has an RMSE of 142.6 points, MAE of 93.4 units, and MAPE of only 5.8%, demonstrating the model's high sensitivity and adaptability to short-term fluctuations in inventory. When the forecast horizon was extended to 60 days, the RMSE increased to 187.2 units and the MAPE to 7.4%. This is mainly due to an increase in seasonal and promotional fluctuations, combined with a loss of information due to prolonged time dependencies. In the 90-day forecast, the RMSE was 231.5 units and the MAE was 162.7 units, with an error increase limited to 35%. This demonstrates that the LSTM architecture remains robust in detecting long-term trends. Overall, the model exhibits minimal error fluctuations across all forecast horizons, demonstrating strong generalization ability and trend tracking performance.

5.3 Model Comparison

To validate LSTM's effectiveness in retail inventory forecasting, comparative experiments were conducted using four models: ARIMA, SVR, GRU, and Transformer. The same dataset and parameter configurations were applied across all models. Results are presented in Table 3.

Table 3. Performance Comparison of Different Forecasting Models

Model	RMSE (Units)	MAE (Units)	MAPE (%)
ARIMA	268.3	191.5	10.4
SVR	243.7	172.9	9.7
GRU	198.4	138.2	8.1
Transformer	184.9	129.5	7.8
LSTM	176.5	121.3	7.2

Table 3 shows that the traditional statistical ARIMA model has limited capabilities to capture non-linear characteristics, with an RMSE of 268.3 units and a MAPE of 10.4%. Although the SVR model performs slightly better for short-term forecasts, its errors accumulate rapidly over time, resulting in a MAPE of 9.7%. On the other hand, the GRU model improves time-dependent modeling through a gating mechanism, reducing the RMSE to 198.4 units. However, the LSTM model maintains a clear advantage in the area of long-term forecasts, achieving an RMSE of 176.4 units and a MAPE of 7.2%, which is approximately 34% less error compared to the ARIMA model. The Transformer model showed a slight improvement in detecting sudden fluctuations, but its generalization stability was worse than that of LSTM due to its sensitivity to sample size and parameters. Detailed comparisons show that LSTM achieves an optimal balance between temporary memory, long-term trend correction, and inventory fluctuation modeling, making it suitable for complex retail business environments.

5.4 Error Analysis

To analyze the stability of the LSTM model's predictions, we performed a statistical distribution analysis on the residuals of the test set, as shown in Figure 2. The residual values are mainly concentrated in the interval [-200, 200] and show an approximately symmetric distribution, with a mean of nearly zero. This indicates that there is no systematic bias in the model. Approximately 94% of the samples show an absolute prediction error of less than 150 units and are mainly concentrated in the interval [-80, 100]. This indicates that the model accurately follows the fundamental trend of the stock.

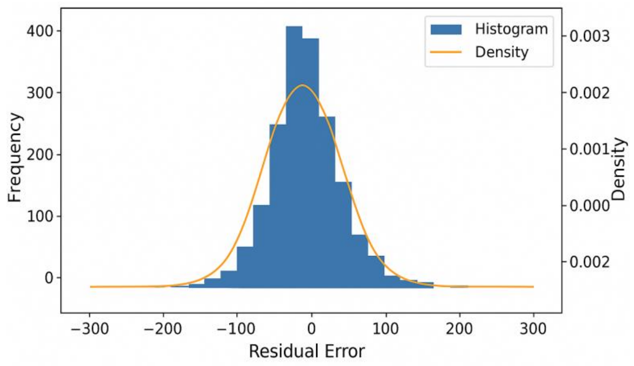

FIGURE 2. ERROR RESIDUAL DISTRIBUTION PLOT

The few outliers in Figure 2 correspond to periods of sharp inventory fluctuations during holidays and promotions. These spikes resulted from sudden replenishments and sales peaks, amplifying short-term prediction errors. Overall, the LSTM model demonstrates high accuracy during stable periods but exhibits slight lag in responding to sudden demand changes. Future improvements could incorporate attention mechanisms or external market signals to enhance anomaly detection capabilities, thereby further reducing residual dispersion.

6 CONCLUSION

The results demonstrate that the LSTM-based deep learning model effectively captures the long-term temporal characteristics and nonlinear fluctuation relationships of retail inventory, significantly improving prediction accuracy and stability. The model exhibits strong generalization and trend-tracking capabilities across multiple forecasting horizons, providing quantitative support for dynamic inventory management. Future work may further incorporate attention mechanisms and multi-source external data to enhance responsiveness to sudden demand changes, enabling the construction of a higher-dimensional intelligent inventory forecasting system for retail operations.

ACKNOWLEDGMENTS

The authors thank the editor and anonymous reviewers for their helpful comments and valuable suggestions.

FUNDING

Not applicable.

INSTITUTIONAL REVIEW BOARD STATEMENT

Not applicable.

INFORMED CONSENT STATEMENT

Not applicable.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

CONFLICT OF INTEREST

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

PUBLISHER'S NOTE

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

AUTHOR CONTRIBUTIONS

Not applicable.

ABOUT THE AUTHORS

HUANG, Sichong

Duke University, USA.

REFERENCES

[1] Wang, C., & Wang, J. (2025). Research on e-commerce inventory sales forecasting model based on ARIMA and LSTM algorithm. Mathematics, 13(11), 1838.

[2] Rong, L., & Vinay, V. (2024). Optimizing supply chain management through BO-CNN-LSTM for demand forecasting and inventory management. Journal of Organizational and End User Computing (JOEUC), 36(1), 1–25.

[3] Ming, Y. T., Yin, K. C., Yuiyip, L., et al. (2023). Data-intensive inventory forecasting with artificial intelligence models for cross-border e-commerce service automation. Applied Sciences, 13(5), 3051.

[4] Yang, H., & Yu, L. (2023). A method of forecasting cross-border e-commerce stocking for SMEs based on demand characteristics and sequence trends under sustainable development strategy. International Journal of Computational Systems Engineering, 7(2–4), 57–66.

[5] Myungsoo, K., Jaehyeong, L., Chaegyu, L., et al. (2022). Framework of 2D KDE and LSTM-based forecasting for cost-effective inventory management in smart manufacturing. Applied Sciences, 12(5), 2380.