Journal of Engineering, Project, and Production Management
2026, 16(5), 2025-366

Multi-Node Supply Chain Demand Forecasting and Fluctuation Modeling Based on Transformer-LSTM Hybrid Model

Jing Feng
Lecturer, School of Economics and Management, Tianjin Vocational Institute, Tianjin, 300341, China, E-mail: jingdaihuakai138@163.com

Project and Production Management
Received December 29, 2025; revised March 4, 2026; accepted April 7, 2026
Available online June 17, 2026

Abstract: Existing research has limitations in co-modeling and quantifying uncertainties in demand sequences across multiple nodes in a supply chain, particularly in addressing local temporal patterns and global dynamic correlations. This paper proposes a Transformer-Long Short-Term Memory (Transformer-LSTM) hybrid model based on a dual-attention gated fusion architecture, denoted as Dual-Attention Gated Fusion Hybrid Model (DGFM). This model extracts local dependency features and global correlation features of the sequence through a parallel-running Long Short-Term Memory (LSTM) network and a Transformer encoder, respectively. A learnable gating weight matrix is applied to adaptively weight and fuse these two types of features, generating a hybrid representation with multi-scale spatiotemporal awareness. Using a quantile regression output layer, the representation generates a nonparametric conditional probability distribution that effectively captures demand fluctuations. The experimental results indicate that the global feature weight increases to 0.85 during promotional periods. The correlation coefficients of the DGFM model remain below 0.31 across all node pairs. This research provides a useful quantitative analysis tool for understanding demand perception and risk-taking in such complex supply chains.

Keywords: Supply chain demand forecasting, Transformer-LSTM hybrid model, volatility modeling; quantile regression, multi-node time series analysis.

DOI 10.32738/JEPPM-2025-366

1. Introduction

Globalization has prompted supply chains to take on characteristics typical of multiple nodes, long cycles, and high dynamism. Understanding demand signals and uncertainty is vital for inventory optimization, production planning, and risk management (Niu et al., 2024; Lang et al., 2024; Lorente-Leyva et al., 2024). The emergence of a global fast fashion phenomenon and greater volatility in market demand has made it more difficult for traditional forecasting models dependent on historical average data to satisfy decision making needs in the textile supply chain (Koren and Shnaiderman, 2023; Kačmáry and Lörinc, 2023). Developing an intelligent forecasting framework based on high-order time-series modeling has emerged as a promising approach to enhance the resilience of the industrial chain.

The key issues with existing research are reconstructing and collaboratively modeling multidimensional time-series patterns. At the data level, multi-node demand sequences simultaneously contains short-term seasonal fluctuations, medium- and long-term trend evolutions, and sudden event disturbances, by combining their own autocorrelation (temporal dependencies within a particular node) and the correlation between the dynamics of that node with those of other nodes (both synchronous and nonsynchronous relationships among different nodes), as well as the response of all nodes to a common external factor (e.g., a market shock, a promotional event) (Hao and Liu, 2024; Shen et al., 2024). At the model level, single time series prediction architectures have inherent limitations. Although models based on recurrent neural networks are good at capturing local dependencies, their sequence recursion mechanism is prone to long-range gradient decay, making it difficult to model macroscopic patterns spanning multiple cycles. Although models based on self-attention mechanisms can directly establish global correlations, there are theoretical boundaries to the perception accuracy of local fine-grained time series patterns (Zhang et al., 2024; Cao et al., 2024). Existing hybrid models adopt simple cascaded or parallel structures, lacking adaptive fusion mechanisms for heterogeneous features, and most studies focus on improving point prediction

accuracy, failing to use volatility as an explicit modeling objective for joint optimization (Rezki and Mansouri, 2024; Zhang et al., 2025).

In response to the above academic difficulties, existing research has attempted breakthroughs through various approaches. The Long Short-Term Memory (LSTM) network and their variants have mitigated gradient vanishing through gating mechanisms, achieving some success in short-term textile demand forecasting. However, their memory unit capacity limits the storage and reuse of complex global contexts (Wang et al., 2023; Prater et al., 2024). Transformer models use self-attention weights to reconstruct the global representation of sequences, thereby providing a new paradigm for modeling long-range dependencies. However, their inductive bias towards local continuity of sequences is weak, easily leading to lag bias at inflection points in demand sequences (Zeng et al., 2022; Oliveira and Ramos, 2024). Probabilistic prediction methods such as quantile regression forests and deep ARIMA models extend the output from a scalar to a distribution, providing insights into volatility estimation. However, these methods typically rely on strong parameter assumptions or are difficult to integrate efficiently with deep representation learning frameworks (Rügamer et al., 2021; Ruiz-Abellón et al., 2024). Overall, existing methods have not yet achieved organic synergy between local temporal patterns and global dynamic correlations and lack end-to-end volatility quantification capabilities.

To address the challenges of collaborative modeling of local and global spatiotemporal patterns and joint uncertainty quantification in multi-node supply chain demand sequences, this paper designs a dual-attention gated fusion model. This method exploits a bidirectional LSTM network to abstract local bidirectional dependency features of the sequence, while simultaneously employing a multi-layer Transformer encoder to capture global spatiotemporal correlations across nodes. Next, a gated feature fusion module uses a learnable weight matrix to dynamically weight and combine the two heterogeneous feature representations. Finally, the fused multi-scale representation is simultaneously fed into both the point prediction branch and the quantile regression branch, achieving synchronous outputs of the demand mean and the conditional probability distribution. This research constructs a complete framework for textile supply chain demand analysis, establishes a technical path for adaptive fusion of multi-scale temporal features, and promotes a paradigm shift in supply chain forecasting from deterministic estimation to probabilistic perception.

2. Construction of DGFM Hybrid Model Based on Transformer-LSTM

2.1. Preprocessing and Input Representation of Multi-Node Time Series Data

The quality and representation of the model input data directly affect the efficiency of subsequent feature extraction and fusion. This study deals with multivariate time series of a multi-node supply chain, whose original data suffer from dimensional differences and local missing values. To address these issues, the min-max normalization method is used to scale the feature values to the [0,1] interval (Tawakuli et al., 2024; Singh and Singh, 2022; Park et al., 2023), as shown in Eq. (1). For randomly missing values, the forward imputation method corresponding to the node is used to interpolate, ensuring the continuity of the time series.

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{1}$$

As shown in Eq. (1) X represents the original feature value, X_{min} and X_{max} are the minimum and maximum values of that feature on the training set, respectively, and X_{norm} is the normalized result. This process eliminates the interference of different physical dimensions on model training.

To construct a tensor input that the model can recognize, the preprocessed sequence data is integrated into a three-dimensional tensor $X \in R^{N \times T \times F}$. The physical meanings of the three dimensions of this tensor are: the number of samples N in the batch, the backtracking time step T, and the feature dimension F. The feature dimension F covers the target variable, time series covariates, market external variables, and event variables. The detailed definitions and preprocessing methods of these features are shown in Table 1.

Table 1. Input feature description and preprocessing methods

Feature category	Feature name	Data source	Preprocessing methods	Physical meaning
Target variable	Historical demand	Internal ERP (Enterprise Resource Planning) system	Maximum and minimum normalization	Core prediction targets of each node
Timing characteristics	Monthly cycle	Timestamp derivation	Sine-cosine encoding	Capturing annual seasonal patterns
Market characteristics	Raw materials price index	Public market data	Z-score normalization	Reflecting upstream cost fluctuations
Event characteristics	Promotional activity logo	Marketing plan	One-hot encoding	Mark abnormal demand fluctuation points

Promotional and event variables: The promotional activity variable is coded as a binary (1=promotion active, 0=promotion not active) to indicate whether a promotion was active during the week being predicted. In addition, a feature has been added to capture the effect of promotion duration by counting the number of consecutive weeks the promotion has been running, capping at 4. The features for the promotional activity were derived from the company's forward-looking marketing plans and could be determined prior to making the forecast (t−1) to avoid lookahead bias. Event variables were created using a one-hot encoding technique based on historical records of the event and can be determined prior to making the forecast (t−1), with no lookahead bias.

The absolute positions and relative intervals of time series are crucial for dependency modeling. The Transformer architecture itself does not have time-series awareness, so positional encoding needs to be injected into the input tensor. This study uses learnable positional encoding, which assigns an independent trainable vector $P_t \in R^F$ to each time step t. The output E of the final embedding layer is obtained by element-wise addition of the normalized input tensor and the positional encoding, as shown in Eq. (2).

$$E_{i,t,f} = \mathcal{X}_{i,t,f} + P_{t,f} \tag{2}$$

In Eq. (2) i is the node index, t is the time step index, f is the feature index. This operation enables the model to explicitly perceive the order and position of data on the time axis, providing necessary structural information for subsequent attention mechanism calculations. After the above preprocessing and embedding, the original business data is transformed into a high-dimensional tensor representation suitable for deep learning.

2.2. Parallel Extraction of Local Dependencies and Global Associations

To collaboratively capture local temporal dependencies and global dynamic correlations in the supply chain demand sequence, this module adopts a dual-path parallel architecture, which uses a long short-term memory network and a self-attention mechanism to extract features from the input embedding tensor $E \in R^{N\times T\times F}$. The overall structure is shown in Fig.1.

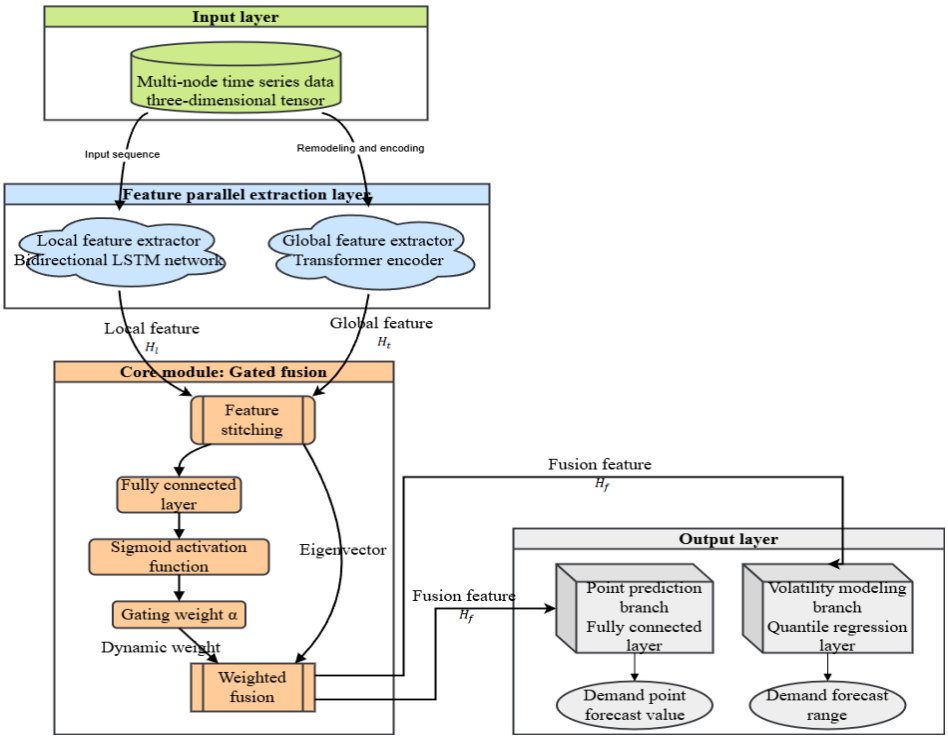

Fig. 1. DGFM model architecture diagram

In the local feature extraction pathway, a bidirectional LSTM network is used as the core encoder. This network achieves refined modeling of short-term temporal patterns through the collaborative mechanism of the input gate i_t, forget gate f_t, output gate o_t, and candidate memory units $\tilde{c}_t$ (Malashin et al., 2024; Da Silva et al., 2023). For the input sequence E_n of the n-th node, the hidden states of its forward and backward LSTM are updated step by step, and the calculation process is defined by Eqs. (3) to (7).

$$i_t = \sigma\left(W_{ii}E_{n,t} + b_{ii} + W_{hi}h_{t-1} + b_{hi}\right) \tag{3}$$

$$f_t = \sigma\left(W_{if}E_{n,t} + b_{if} + W_{hf}h_{t-1} + b_{hf}\right) \tag{4}$$

$$o_t = \sigma\left(W_{io}E_{n,t} + b_{io} + W_{ho}h_{t-1} + b_{ho}\right) \tag{5}$$

$$\tilde{c}_t = \tanh\left(W_{ig}E_{n,t} + b_{ig} + W_{hg}h_{t-1} + b_{hg}\right) \tag{6}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{7}$$

σ represents the sigmoid activation function; ⊙ represents the Hadamard product, W and b are trainable parameters (Kılıçarslan et al., 2021; Yuen et al., 2021). Finally, the concatenation $[\overrightarrow{h_T}, \overleftarrow{h_T}]$ of the forward and backward hidden states at the last time step is taken as the local feature representation H_l^n of that node. The local features of all nodes constitute a set $H_l \in R^{N \times D_h}$, where D_h is the hidden layer dimension.

In the global feature extraction pathway, a multi-layer Transformer encoder is used to reconstruct the global context representation of the sequence (Shusen et al., 2023; Deihim et al., 2023). First, the input tensor E is reshaped into a two-dimensional matrix $E' \in R^{(N \cdot T) \times F}$ to compute multi-head self-attention. The computation of each attention head is defined in Eq. (8).

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{8}$$

In Eq. (8), the query matrix Q, key matrix K, and value matrix V are obtained by linear projection of the input E', and d_k is the key vector dimension. The output of the multi-head attention is processed by layer normalization and a feedforward neural network and then aggregated into a fixed-dimensional global feature vector $H_t \in R^{D_h}$ through a global average pooling operation. This vector compresses the long-range spatiotemporal correlation pattern across nodes and time steps (Dogan, 2023; Zhao and Zhang, 2024). The outputs of the two paths H_l and H_t are used together as the input of the gating fusion module.

2.3. Gated Feature Fusion and Probability Output Layer

The local feature representation $H_l \in \mathbb{R}^{N \times D_h}$ generated by the module and the global feature representation $H_t \in \mathbb{R}^{D_h}$ exhibit semantic and scale heterogeneity. To achieve organic synergy between the two, this module applies to a gated fusion mechanism. This mechanism first broadcasts the global feature vector H_t to align its dimension with H_l, resulting in $H_t' \in \mathbb{R}^{N \times D_h}$. Subsequently, H_l and H_t' are concatenated along the feature dimension to form a joint feature representation $H_c = [H_l; H_t'] \in \mathbb{R}^{N \times 2D_h}$. This joint feature is mapped through a fully connected layer and a Sigmoid activation function to generate a gated weight vector α with the same dimension as the original feature, which is expressed in Eq. (9).

$$\alpha = \sigma(W_g H_c + b_g) \tag{9}$$

In Eq. (9), $W_g \in R^{D_h \times 2D_h}$ and $b_g \in R^{D_h}$ are trainable parameters, and σ is the Sigmoid function. The value range of each element of this weight vector α is [0,1], which is used to dynamically adjust the contribution of the two types of features. The final fused features $H_f \in R^{N \times D_h}$ are obtained by gated weighted sum, calculated as shown in Eq. (10).

$$H_f = \alpha \odot H_l + (1 - \alpha) \odot H_t' \tag{10}$$

This design enables the model to adaptively select between relying on local time-series patterns or global context information based on the inherent characteristics of the input data. To simultaneously accomplish demand point prediction and volatility modeling, the fused features H_f are input into a two-branch output layer. The point prediction branch is a linear transformation layer that can produce the output demand forecast value $\hat{y} \in \mathbb{R}^N$ for the desired future time step, expressed in Eq. (11).

$$\hat{y} = W_p H_f + b_p \tag{11}$$

$W_p \in \mathbb{R}^{1 \times D_h}$ and $b_p \in \mathbb{R}$ are parameters of the point prediction branch. The fluctuation modeling branch employs a quantile regression approach, producing predicted values for several target quantiles via a collection of independent linear transformation layers (Wang et al., 2024; Yang et al., 2024).

During training, all parameters are optimized jointly by minimizing the quantile loss function L_τ, which is defined as the weighted absolute deviations between the predicted and true values, as specified in Eq. (12).

$$L_\tau = \frac{1}{N}\sum_{i=1}^{N} \max\left(\tau(y_i - \hat{y}_{\tau,i}), (1 - \tau)(\hat{y}_{\tau,i} - y_i)\right) \tag{12}$$

By producing predicted values of several key quantiles, the model can represent a nonparametric form of the conditional prediction distribution and thus quantify the uncertainty of demand.

3. Experimental Setup and Verification

3.1. Data Collection and Experimental Environment

To evaluate the performance of the proposed model in an industrial context, the analysis utilizes over 36 months of real supply chain operation data from a large textile and apparel group. The dataset contains comprehensive information about the four primary supply chain components and their normal processes (textile raw materials, yarn, grey fabric and finished garments). Furthermore, it provides an accurate portrayal of demand transmission characteristics and demand variation in a multilevel production and distribution environment. Demand data for each node is recorded in chronological order over weekly intervals, as are the time series, market and operational features. The dataset will be partitioned into three non-overlapping subsets in chronological order. The first 70% of the data sequence will form the training dataset for learning

model parameters. The next 10% of the sequence will serve as the validation dataset for selecting hyperparameters and for determining early stopping criteria. The final 20% of the sequence will function as the test dataset for evaluating final performance and conducting comparative analyses. This dataset partitioning exercise follows the premise of time-series forecasting and guards against any potential data traversal issues.

To provide a comprehensive understanding of the dataset, we further detail its key features. The dataset contains weekly records for each node, with approximately 156 samples per node. However, due to data collection gaps, the actual numbers are raw materials 152, yarn 154, greige fabric 150, and finished garments 155. The missing value rate is low, ranging from 1.3% to 3.8% across nodes. Missing values were imputed using forward imputation. To assess the impact of imputation, we conducted a sensitivity analysis, comparing model performance on the complete subset. The mean absolute error difference was less than 2%, indicating a negligible impact. Promotional events were defined as periods with at least a 15% discount on finished garments, and this agreement was derived from the company's marketing plan database. Other events included supplier disruptions or sudden order changes, tagged using one-hot encoding. All features were available at prediction time, namely historical demand up to week t-1, known future promotions, and market indices released one week later, ensuring no look-ahead bias in the experimental setup.

All experiments run on servers with NVIDIA GPUs, which have video memory that can accommodate scaling tensor computations during model training. The programming environment is based on Python 3.8 and the PyTorch 1.12.1 deep learning framework for flexibility in model deployment and training. This requires the models to be trained with the AdamW optimizer and an initial learning rate of 1e-3, adjusted using cosine annealing (Zhang et al., 2024; Arthur et al., 2024). The batch size for training is 32, and the model training is completed in 300 epochs. Additionally, an early stopping mechanism is used, where training takes place for 10 epochs without improvement of the validation loss on the validation set to avoid overfitting (Li et al., 2024; Sabiri et al., 2022). To ensure the reliability of the experimental results, all comparative experiments are repeated 5 times under the same hardware and software environment and data partitioning, and their average performance metrics are reported.

3.2. Evaluation Indicators and Comparison Scheme

To fully quantify the predictive performance of the model, two types of evaluation metrics are used in the experiment. Point prediction accuracy is measured by mean absolute error. Mean absolute error is calculated as the average of the absolute deviations between the predicted value and the true value. It is insensitive to outliers and provides a robust accuracy estimate. Fluctuation prediction reliability is evaluated by mean coverage error and mean interval width (Nikulchev and Chervyakov, 2023). Mean coverage measures the proportion of the true value falling within the prediction interval of a specified confidence level. Its deviation from the theoretical confidence level is calculated by the mean coverage error. The construction of the comparison scheme aims to verify the effectiveness of the model in this paper from the perspective of methodological evolution. The set of benchmark models covers representative methods from traditional time series models to innovative deep learning architectures. The specific model composition and key parameter configuration are shown in Table 2.

All baseline models utilized the same input feature set as that of the DGFM models to permit a fair comparison between the two sets of models. The hyperparameters for each model had been tuned on the validation set through a grid search. The same optimizer, learning rate schedule, batch size, and early stopping criteria were used when training each of the models in order for all models to be trained with comparable computational budgets. Predictive intervals for the probabilistic models (e.g., DeepAR) were created using their respective internal likelihood functions, while for the deterministic models, interval predictions were made by training a quantile regression output layer to maintain consistent volatility modeling.

The spectrum of technical models, as illustrated in Table 2, shows both a continuum from traditional to modern as well as from simple to integrated. The established baseline models for comparative purposes are classical statistical modeling and machine-learning models. The proposed deep sequence models, LSTM and Transformer, represent the two most common methods used for capturing local dependencies and global correlations. The difference in performance between DGFM and the benchmark models is the result of the success of its gated fusion architecture at concurrently modeling multi-scaled temporal characteristics.

4. Results

4.1. Comparison of Point Prediction Accuracy

The accuracy of point predictions is an essential measure of how well a demand forecasting model performs and directly impacts how trustworthy the master supply chain planning process is when relied upon for execution. This section will demonstrate how the dual attention gate fusion model (DAGFM) outperforms traditional demand mean estimators through a comparative analysis of the Mean Absolute Errors (MAE) of five different types of models (ARIMA, XGBoost, LSTM, Transformer, and DAGFM producing demand forecasts against a complete dataset for each of the four nodes across the supply chain, namely raw material, yarn, greige, and finished garments. Fig. 2 shows the results from this analysis.

All baseline models used the same input feature set as that of the DGFM models to permit a fair comparison between the two sets of models. The hyperparameters for each model were tuned on the validation set using a grid search. The same optimizer, learning rate schedule, batch size, and early stopping criteria were used when training each model, ensuring comparable computational budgets across all models. Predictive intervals for probabilistic models (e.g., DeepAR) were computed using their respective internal likelihood functions, whereas for deterministic models, interval predictions were obtained by training a quantile regression output layer to maintain consistent volatility modeling.

The spectrum of technical models, as illustrated in Table 2, shows both a continuum from traditional to modern as well as from simple to integrated. The established baseline models for comparative purposes are classical statistical modeling and machine-learning modeling. The proposed deep sequence models, LSTM and Transformer, represent the two most common methods used for capturing local dependencies and global correlations. The difference in performance between DGFM and the benchmark models is due to its gated fusion architecture, which concurrently models multi-scale temporal characteristics.

5. Results

5.1. Comparison of Point Prediction Accuracy

The accuracy of point predictions is an essential measure of how well a demand forecasting model performs and directly impacts how trustworthy the master supply chain planning process is when relied upon for execution. This section will demonstrate how the DAGFM outperforms traditional demand mean estimators through a comparative analysis of the Mean Absolute Errors (MAE) of five different types of models (ARIMA, XGBoost, LSTM, Transformer, and DAGFM) producing demand forecasts against a complete dataset for each of the four nodes across the supply chain; namely raw material, yarn, greige, and finished garments. Fig. 2 shows the results from this analysis.

Table 2. Comparison of models and parameter settings

Model category	Model name	Core parameter settings	Theoretical basis
Traditional statistical model	ARIMA	(p, d, q) are automatically determined by the AIC (Akaike's Information Criterion) criterion	Autoregressive integrated moving average classic benchmark
Machine learning model	XGBoost	Max depth=6, n_estimators=100, learning rate=0.1	Gradient boosting decision tree, powerful feature interaction capabilities
Deep learning sequence model	LSTM	2-layer stacking, 64 hidden units, Dropout=0.2	Recurrent neural networks are good at capturing short-term temporal dependencies
Deep learning sequence model	Transformer	4 attention heads, 2 encoder layers, feedforward dimension 128	Based on the self-attention mechanism, it is good at capturing global long-term dependencies
This paper's model	DGFM	Hidden dimension 128, gated fusion layer dimension 256	Adaptive fusion of local and global features
Deep learning probabilistic models	DeepAR	2-layer LSTM, 64 hidden cells, Gaussian likelihood	Autoregressive probability prediction
Advanced transformer variants	Informer	Attention head 4, encoder layer 2, ProbSparse self-attention	Long sequence efficient prediction model

5.2. Reliability Assessment of Fluctuation Modeling

Accurate demand forecasting still requires effective quantification of uncertainty to support supply chain risk management decisions. Volatility modeling aims to depict the probability of distribution of future demand, and its reliability directly determines the scientific validity of safety stock settings and service level commitments. The model's performance in quantifying demand uncertainty is assessed by comparing forecast interval coverage with the actual value and by examining the characteristics of the forecast interval widths (as shown in Fig 3).

As illustrated in Fig. 3 above, the x-axis represents a time series of the test data set, and the y-axis represents the level of market demand expressed in thousands of garments. The DGFM model's empirical prediction interval at a 95% confidence level is being compared with the actual historical demand sequence. During the promotional/peak phase of the season (time series 3-5), actual demand increased sharply to 168.9 thousand garments. The lower limit of the prediction interval (128.5 thousand garments) and the upper limit (185.9 thousand garments) fully cover this dramatic increase in demand during this period, with the difference between the upper and lower limits remaining within an acceptable range of approximately 57

thousand garments. The outstanding prediction results during this phase of the season can be attributed to the Transformer encoder's ability to quickly assess sudden demand changes and to the gating mechanism, which dynamically enhances global characteristics, enabling the model to predict and respond effectively to systemic demand variability. By comparison, in the more stable time series 7, actual demand returned to 88.7 thousand garments, and the prediction interval narrowed to approximately 32 thousand garments, indicating the LSTM network's consistency in estimating a baseline demand level. The average interval width throughout the testing period indicates that the model avoids overly conservative intervals while maintaining high coverage. This balance is achieved through the joint optimization of interval width and coverage by the quantile regression loss. The DGFM model accurately quantifies demand fluctuations via the probability output layer, providing a reliable basis for decision-making in supply chain risk management.

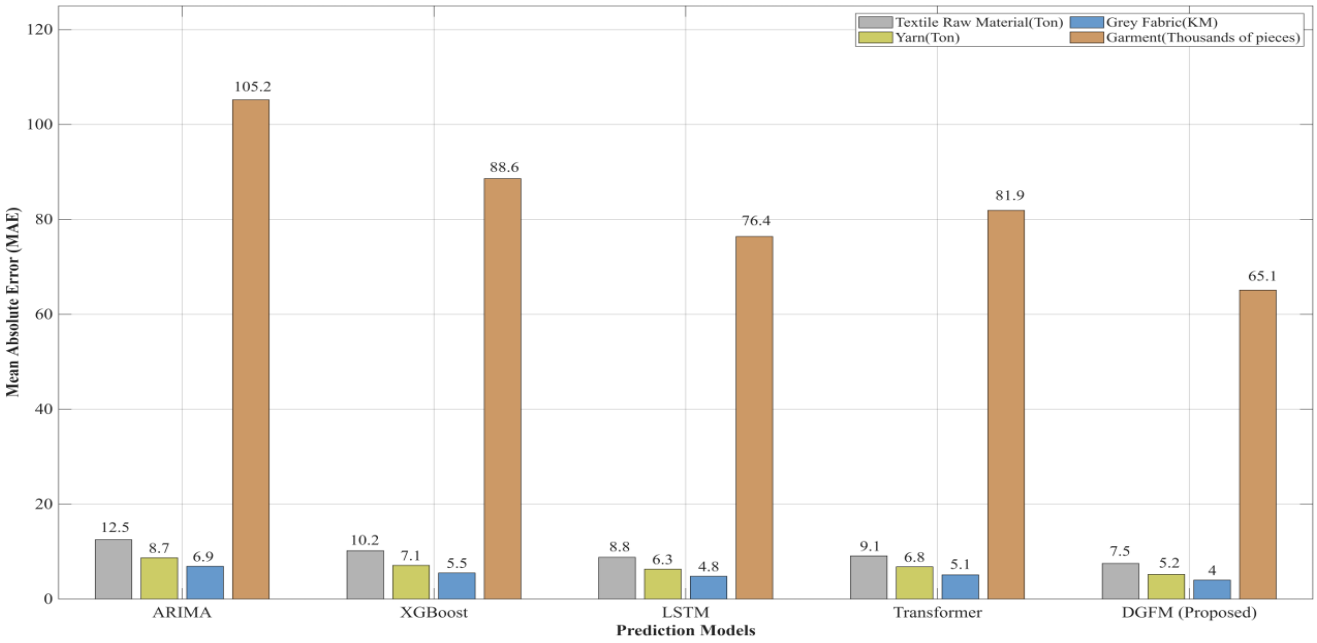

Fig. 2. Comparison of MAE across multiple supply chain nodes

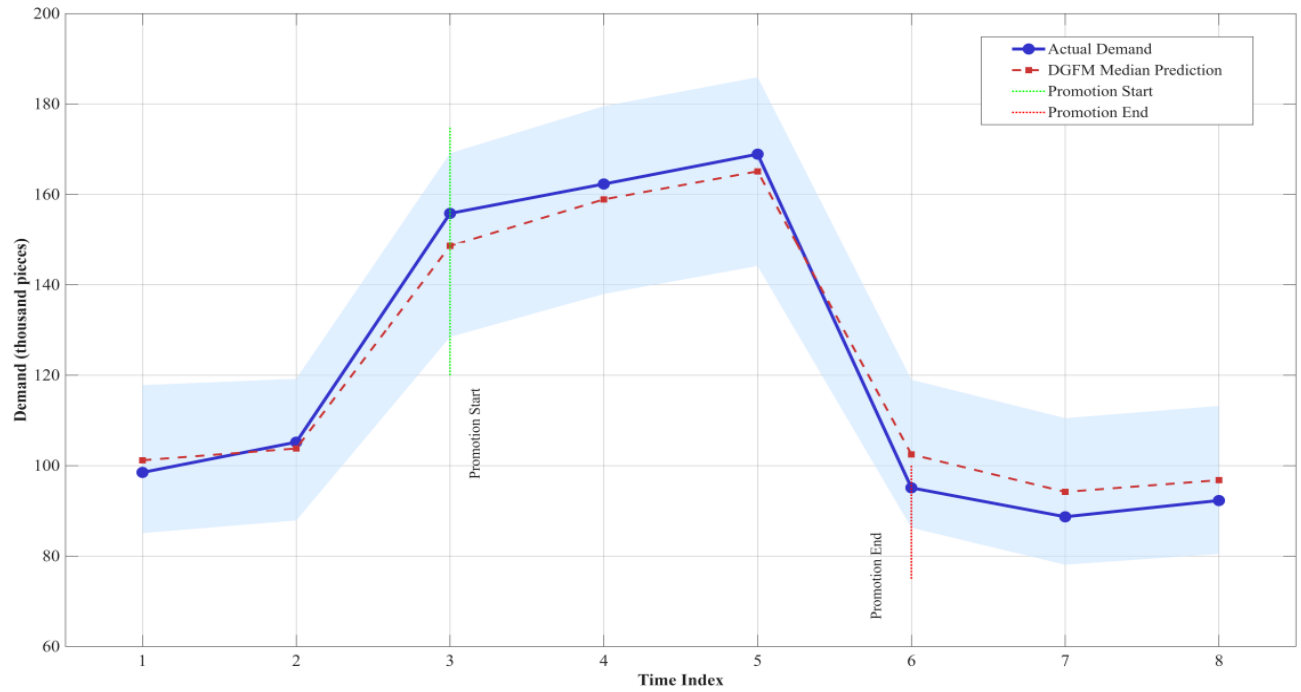

Fig. 3. Visualization of predicted interval coverage

5.3. Multi-Node Prediction Consistency

Because multiple nodes in the supply chain have interrelated demand effects, forecasting models need accurate forecasting at a minimum in a single-point sense, but also in the sense of systemic collaborative forecasting models. The high correlation

in forecasting errors across nodes is evidence of inadequate decoupling of complex supply-and-demand relationships, which can lead to the accumulation and escalation of forecasting errors throughout the supply chain. This section evaluates the global collaborative forecasting effectiveness by calculating and comparing the correlation coefficients of forecasting errors across node pairs for different models, as shown in Fig. 4.

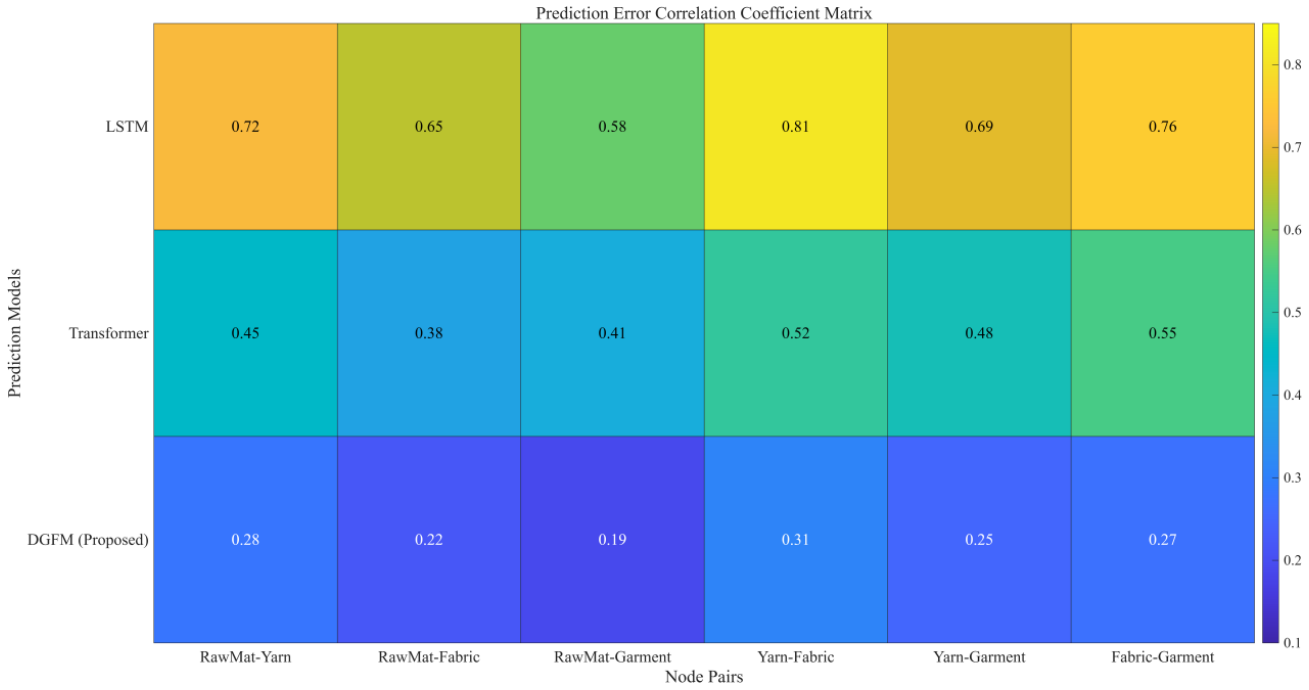

Fig. 4. Correlation coefficient matrix of prediction errors

Fig. 4 shows the three deep learning models on the vertical axis and the six supply chain node pairs on the horizontal axis. The color intensity represents the Pearson correlation coefficient of the prediction error. The LSTM model's error correlation is mostly higher than 0.65, reaching 0.81 for the yarn-grey fabric node pair, indicating that its mechanism, based on local sequence modeling, struggles to distinguish the mutual influence between nodes. The Transformer model reduces the correlation to the range of 0.38 to 0.55 through a self-attention mechanism, demonstrating the effectiveness of global modeling in reducing error propagation. The DGFM model in this paper has a correlation coefficient of no more than 0.31 for all node pairs, with the correlation for the raw material-garment node pair being only 0.19. This performance is attributed to the gating fusion mechanism's enhancement of node-specific features and suppression of global interference factors. The low and uniform error correlation shows that the model can provide more stable system-level prediction capabilities, obviating risk associated with the bullwhip effect through collaborative learning in the supply chain.

5.4. Verification of the Effectiveness of the Gating Mechanism

The gating fusion module is the core component of the model in this paper to enable adaptive interaction between local and global features. The effectiveness of the dynamic decision-making process directly determines the model's capability to respond to complex demand patterns. In this regard, this section addresses the evolution of the gating weight α across various operational conditions and analyzes the model's strategy for allocating weights during the testing period for the garment node. The evolution of α over time displays a dynamic response process, presented in Fig. 5.

Fig. 5 shows the test set time series on the horizontal axis and the gating weights α on the vertical axis, physically representing the model's dependence on the global features of the Transformer. During the stable operation phase (time series 1-2), the α value remains stable around 0.32 to 0.35, indicating that the model relies on the LSTM network to capture local temporal patterns. When the time series progresses to point three, the promotional activity triggers a sudden demand surge, and the α value rapidly jumps to 0.78. The sharp change is due to the Transformer encoder's global contextual understanding of abrupt changes, which allows the model to recognize changing patterns in systemic demand more quickly than before. In the peak season period series 4-5, "α" increases further to 0.85 (the peak), indicating that the model continues to use global features and builds on them to accommodate high demand fluctuations. After the promotional event has passed (time series 6), the "α" returned to 0.65 yet remains above the stable range levels, indicating that the model has retained some level of memory with respect to global data to help mitigate the inertial impact of demand decreases. The weight allocation model is extremely applicable to real-life business situations; therefore, it further validates the gating function's ability to independently adjust the relative proportions of local and global data contributed to the merger of local and global feature locations based on the natural characteristics of the input variables, thus enabling true adaptive integration.

6. Conclusion

In this research project, we create a dual-attention gated fusion model to address the challenges of multi-node demand forecasting and volatility modeling in textile supply chains. We will present three main contributions. 1) a parallel

architecture that simultaneously extracts local temporal features with a Bi-directional Long Short-Term Memory (BiLSTM) incremental feature extractor, and global correlation features with a Transformer (multi-head self-attention component). 2) an innovative gated fusion mechanism that quantitatively weights and synthesizes the two different feature types based on the characteristics of the input data. 3) the use of a quantile regression output layer to optimize both point forecasts and prediction intervals in an end-to-end manner, allowing us to quantify uncertainty through our entire system. Evidence from experiments suggests the model's most accurate single selection points are four supply chain nodes, raw materials, yarn, greige fabric and finished garments, with the greatest MAE at the grey garment node decreased to 65100 pieces. The gating weight analysis indicates that the feature fusion method is flexible enough to alter weights for selected features based on contextual changes in operations, and especially during promotional periods, where weights for global features increased to 0.85. Given the findings of this research, it represents a tool for combining demand forecasting with an understanding of risk and uncertainty in textile supply chains, with considerable use value for managers in inventory optimization and risk management decisions within supply chain systems. There remain some limitations for applying this research generally and its CR for the specification of selected features. Future research will advance the development of multimodality and graph-based architecture methodologies to enhance multi-node reliability and consistency.

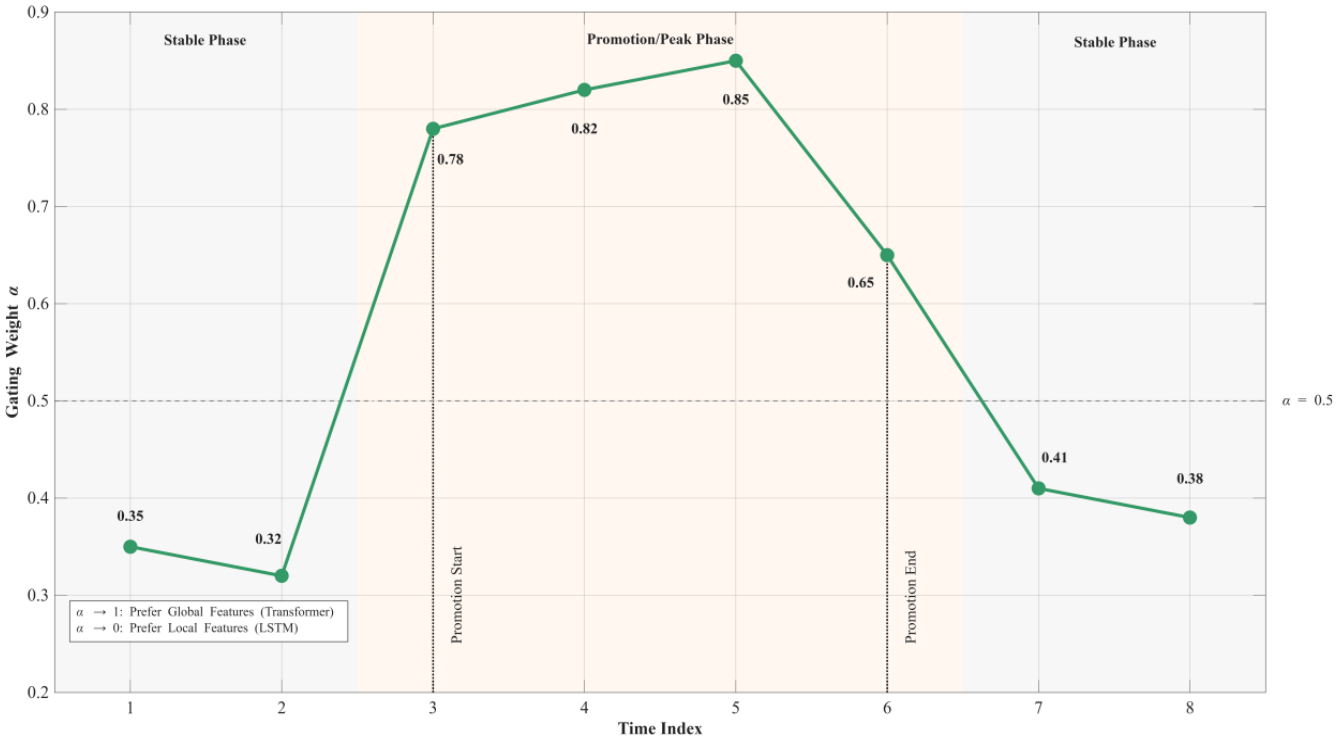

Fig. 5. Temporal evolution of gating weight

Funding

This research was supported by the achievements of the Scientific Research Plan Project of Tianjin Municipal Education Commission (Grant number: 2025SK162). Research on the Identification and Reconstruction Path of the Skill Gap of Logistics Industry Workers Driven by Digital Intelligence.

Institutional Review Board Statement

Not applicable.

Declaration of Artificial Intelligence (AI) Tools

The author used Deepseek solely for language editing and readability improvement. The author reviewed and verified all content and takes full responsibility for the accuracy and integrity of the manuscript.

References

Arthur, C., Yudistira, N., and Dewi, C. (2024). Autocyclic: Deep learning optimizer for time series data prediction. *IEEE Access*, 12, 14014-14026.

Cao, K., Zhang, T., and Huang, J. (2024). Advanced hybrid LSTM-transformer architecture for real-time multi-task prediction in engineering systems. *Scientific Reports,* 14(1), 4890.

Da Silva, D. G. and de Moura Meneses, A. (2023). Comparing long short-term memory (LSTM) and bidirectional LSTM deep neural networks for power consumption prediction. *Energy Reports,* 10, 3315-3334.

Deihim, A., Alonso, E., and Apostolopoulou, D. (2023). STTRE: A spatio-temporal transformer with relative embeddings for multivariate time series forecasting. *Neural Networks,* 168, 549-559.

Dogan, Y. (2023). A new global pooling method for deep neural networks: Global average of top-k max-pooling. *Traitement du Signal,* 40(2), 577-587.

Hao, J. and Liu, F. (2024). Improving long-term multivariate time series forecasting with a seasonal-trend decomposition-based 2-dimensional temporal convolution dense network. *Scientific Reports,* 14(1), 1689.

Kačmáry, P. and Lörinc, N. (2023). Possibilities of sale forecasting textile products with a short life cycle. *Sustainability,* 15(21), 15517.

Kılıçarslan, S., Adem, K., and Çelik, M. (2021). An overview of the activation functions used in deep learning algorithms. *Journal of New Results in Science,* 10(3), 75-88.

Koren, M. and Shnaiderman, M. (2023). Forecasting in the fashion industry: a model for minimising supply-chain costs. *International Journal of Fashion Design, Technology and Education,* 16(3), 308-318.

Lang, Q., Hu, J., and Liu, J. (2024). Impact of cost sharing on quality improvement and profits under uncertain demand: The case of a textile and garment supply chain. *PLOS ONE,* 19(5), e0304578.

Li, H., Rajbahadur, G. K., Lin, D., Bezemer, C. P., and Jiang, Z. M. (2024). Keeping deep learning models in check: A history-based approach to mitigate overfitting. *IEEE Access,* 12, 70676-70689.

Lorente-Leyva, L. L., Alemany, M. M. E., and Peluffo-Ordóñez, D. H. (2024). A conceptual framework for the operations planning of the textile supply chains: Insights for sustainable and smart planning in uncertain and dynamic contexts. *Computers & Industrial Engineering, 187*, 109824.

Malashin, I., Tynchenko, V., Gantimurov, A., Nelyub, V., and Borodulin, A. (2024). Applications of long short-term memory (LSTM) networks in polymeric sciences: A review. *Polymers,* 16(18), 2607.

Nikulchev, E. and Chervyakov, A. (2023). Prediction intervals: A geometric view. Symmetry, 15(4), 781.

Niu, T., Zhang, H., Yan, X., and Miao, Q. (2024). Intricate supply chain demand forecasting based on graph convolution network. *Sustainability*, *16*(21), 9608.

Oliveira, J. M. and Ramos, P. (2024). Evaluating the effectiveness of time series transformers for demand forecasting in retail. *Mathematics,* 12(17), 2728.

Park, J., Müller, J., Arora, B., Faybishenko, B., Pastorello, G., Varadharajan, C., Sahu, R., and Agarwal, D. (2023). Long-term missing value imputation for time series data using deep neural networks. *Neural Computing and Applications*, 35(12), 9071-9091.

Prater, R., Hanne, T., and Dornberger, R. (2024). Generalized performance of LSTM in time-series forecasting. *Applied Artificial Intelligence*, 38(1), 2377510.

Rezki, N. and Mansouri, M. (2024). Deep learning hybrid models for effective supply chain risk management: mitigating uncertainty while enhancing demand prediction. *Acta Logistica,* 11(4), 589-604.

Rügamer, D., Baumann, P. F. M., Kneib, T., and Hothorn, T. (2021). Probabilistic time series forecasts with autoregressive transformation models. *Preprint arXiv,* 21(10), 08248.

Ruiz-Abellón, M. C., Fernández-Jiménez, L. A., Guillamón, A., and Gabaldón, A. (2024). Applications of probabilistic forecasting in demand response. *Applied Sciences,* 14(21), 9716.

Sabiri, B., El Asri, B., and Rhanoui, M. (2022). Mechanism of overfitting avoidance techniques for training deep neural networks, 1, 418-427.

Shen, C., He, Y., and Qin, J. (2024). Robust multi-dimensional time series forecasting. Entropy, 26(1), 92.

Shusen, M., Tianhao, Z., and Bo, Y. Z. (2023). TCLN: A transformer-based Conv-LSTM network for multivariate time series forecasting. *Applied Intelligence,* 53(23), 28401-28417.

Singh, D., and Singh, B. (2022). Feature wise normalization: An effective way of normalizing data. Pattern Recognition, 122, 108307.

Tawakuli, A., Havers, B., Gulisano, V., Kaiser, D., and Engel, T. (2024). Survey: Time-series data preprocessing: A survey and an empirical analysis. *Journal of Engineering Research,* 13(2), 674-711.

Wang, J., Wang, S., Lv, M., and Jiang, H. (2024). Forecasting VaR and ES by using deep quantile regression, GANs-based scenario generation, and heterogeneous market hypothesis. *Financial Innovation,* 10(1), 36.

Wang, W., Shao, J., and Jumahong, H. (2023). Fuzzy inference-based LSTM for long-term time series prediction. *Scientific Reports,* 13(1), 20359.

Yang, L., Bai, H., and Ren, M. (2024). Threshold quantile regression neural network. *Applied Economics Letters, 31(17), 1675-1685.*

Yuen, B., Hoang, M. T., Dong, X., and Lu, T. (2021). Universal activation function for machine learning. *Scientific Reports,* 11(1), 18757.

Zeng, A., Chen, M., Zhang, L., and Xu, Q. (2022). Are transformers effective for time series forecasting? *arXiv e-prints,* 22(05), 13504.

Zhang, C., Shao, Y., Sun, H., Xing, L., Zhao, Q., and Zhang, L. (2024). The WuC-Adam algorithm based on joint improvement of Warmup and cosine annealing algorithms. *Math Biosci Eng,* 21(1), 1270-1285.

Zhang, Y., Wu, R., Dascalu, S. M., and Harris Jr., F. C. (2024). Sparse transformer with local and seasonal adaptation for multivariate time series forecasting. *Scientific Reports,* 14(1), 15909.

Zhang, Y., Zhang, T., and Hu, J. (2025). Forecasting stock market volatility using CNN-BiLSTM-attention model with mixed-frequency data. *Mathematics,* 13(11), 1889.

Zhao, L. and Zhang, Z. (2024). An improved pooling method for convolutional neural networks. *Scientific Reports,* 14(1), 1589.

Jing Fen, Master, lecturer at the School of Economics and Management, Tianjin Vocational Institute. Her research interests include logistics and supply chain management and vocational education.