ISSN 1330-3651 (Print), ISSN 1848-6339 (Online) https://doi.org/10.17559/TV-20250930003034
Received: 30 September 2025; Accepted: 17 December 2025 Original scientific paper

Adaptive Deep Learning Ensemble for Supply Chain Demand Forecasting

Kyoungjong PARK

Abstract: Accurate supply chain demand forecasting is critical for inventory optimization and risk reduction. This study proposes three enhanced forecasting models, specifically Dynamic Weight Fusion-based Uni-Regression Deep Approximate Forecasting (DWF_UDAF), Multi-Head Attention-based UDAF (MHA_UDAF), and a combined Dynamic Weight Fusion-MHA model (DWFMHA_UDAF). These models are built upon Bidirectional Long Short-Term Memory (BiLSTM) and Nonlinear Autoregressive models with exogenous inputs (NARX) to address the limitations of fixed-weight neural-ensemble architectures such as UDAF. By introducing Dynamic Weight Fusion (DWF) and multi-head attention (MHA), the proposed architectures adaptively reflect temporal demand shifts. The proposed models are evaluated through a focused methodological validation using a highly volatile time-series (Store 1) from the Rossmann Store Sales dataset. This specific store was selected as a complex testbed to rigorously assess the adaptive capability of the DWF mechanism under dynamic demand shifts and exogenous influences. Empirical validation on the Rossmann Store Sales dataset, benchmarking ten forecasting models, showed that DWF_UDAF achieved the lowest Mean Absolute Error (MAE = 0.147). In contrast, the statistical ARIMAX baseline recorded the lowest Root Mean Squared Error (RMSE = 0.206). Statistical tests, specifically one-way ANOVA and Tukey's HSD, confirmed that DWF_UDAF outperformed both fixed-weight and attention-based ensemble architectures in MAE ($p < 0.001$). In contrast, MHA-based models exhibited degraded accuracy. This study attributes this performance drop to the structural mismatch between the high data requirements of Multi-Head Attention and the limited sample size of a single store, which led to overfitting and attention weight collapse. This work provides both theoretical and practical evidence that adaptive-fusion mechanisms outperform static structures in complex supply chain forecasting scenarios.

Keywords: dynamic weight fusion; multi-head attention; neural-ensemble architectures; supply chain demand forecasting

1 INTRODUCTION

Demand forecasting in supply chains is fundamental to core decision-making in various operational areas, including inventory optimization, production planning, and risk management [1-4]. Following the COVID-19 pandemic, the volatility of demand has significantly increased, elevating forecasting accuracy from a mere technical task to a critical determinant of business performance [5]. Recent theoretical developments in supply chain management emphasize that static supply chains are increasingly vulnerable to disruptions. Wieland (2021) argues that modern supply chains must evolve into "living" systems capable of adapting to complex and changing environments, rather than merely optimizing for stability [6]. In this context, forecasting models must also transit from static structures to adaptive architectures that can learn from dynamic data patterns. However, traditional statistical approaches, such as ARIMA, are built upon linear assumptions and are only effective in capturing short-term trends. These models tend to underperform when applied to complex nonlinear structures or environments influenced by exogenous factors [7, 8].

In response, researchers have increasingly adopted machine and deep learning-based approaches to better handle the nonlinear and volatile nature of real-world demand patterns. Long Short-Term Memory (LSTM) networks capture long-term dependencies, and their bidirectional variant (BiLSTM) further exploits context in both temporal directions [9]. Nonlinear Autoregressive models with exogenous inputs (NARX) complement this strength by modelling nonlinear effects from external variables [10, 11]. The heterogeneous neural-architecture-based Uni-Regression Deep Approximate Forecasting (UDAF), which integrates BiLSTM and NARX, has demonstrated strong predictive performance. However, its reliance on a fixed weight fusion structure limits its adaptability to temporal variations in prediction accuracy [12].

In practical supply chain environments, demand patterns fluctuate dynamically in response to factors such as promotional campaigns, holidays, and weather. Under these circumstances, an ensemble framework should dynamically adjust fusion weights based on the relative performance of predictive modules at each time point. Smyl [13] demonstrated the feasibility of such real-time adjustments and introduced the idea of dynamic weight fusion, which has since gained traction in the time-series forecasting community. This study extends prior dynamic-weight architectures by introducing a gradient-based adaptive WeightOptimizer and coupling it with a heterogeneous BiLSTM-NARX ensemble that explicitly handles exogenous variables, thereby enabling online adaptation and context-aware module prioritization capabilities not jointly realized in earlier approaches such as homogeneous RNN fusion [13] or frequency-based decomposition [14]. Accordingly, this work constitutes an applied, empirical method-development study with an experimental-comparative design positioned at the intersection of supply chain analytics and deep learning-based time-series forecasting, rather than basic theory, a single case study, or management-science modelling.

Meanwhile, attention mechanisms have gained increasing recognition for their effectiveness in emphasizing critical time points in sequential data, thereby mitigating the issue of information dilution in long input sequences. The Transformer architecture suggested by Vaswani et al. [10] has achieved widespread success across numerous domains, and Lim et al. [15] adapted this structure for time-series forecasting through the Temporal Fusion Transformer. However, attention-based models often require careful design of preprocessing components, including input normalization and embedding strategies. Otherwise, they risk overfitting and performance instability.

This study sets out to achieve three key objectives:

(1) to design a dynamic-weight-fusion-based model that addresses the limitations of fixed weight combinations in UDAF,

(2) to experimentally evaluate the effectiveness of multi-head attention in emphasizing critical time steps, and

(3) to apply the proposed framework to a practical supply chain forecasting problem to derive the most effective architectural alternative.

To this end, this study conducts a comparative evaluation using the Rossmann Store Sales dataset provided by Kaggle, testing ten forecasting models. Statistical tests, including one-way ANOVA and Tukey's HSD, are employed to assess the significance of performance differences across models.

The remainder of this paper is organized as follows. Section 2 is the literature on supply chain demand forecasting. Section 3 defines the suggested model and its methodological framework. Section 4 outlines the experimental setup. Section 5 interprets key findings and addresses the study's limitations. Finally, Section 6 states the concluding directions for future study.

2 LITERATURE REVIEW

Supply chain demand forecasting has traditionally evolved through three main methodological paradigms: statistical, machine learning, and deep learning approaches. Among the most widely adopted statistical models are Autoregressive Integrated Moving Average (ARIMA), Moving Average, and Exponential Smoothing methods. These models demonstrate effective performance when the data exhibit simple seasonality or trend patterns [7]. However, real-world supply chain environments often involve nonlinear dynamics and external factors, which significantly limit the predictive accuracy of these traditional methods [5, 7].Theoretical research suggests that developing advanced data analytics capabilities is essential for enhancing supply chain resilience. Jiang et al. (2024) demonstrated that the combination of big data analytics capability and supply chain integration significantly improves a firm's ability to prepare for and respond to disruptions [16]. Chen et al. (2022) introduced the concept of "supply network resilience learning", demonstrating that exploratory data analytics enables firms to develop capabilities for preventing and recovering from disruptions [17]. This supports the shift towards adaptive, data-driven forecasting architectures capable of learning from dynamic environments. These studies provide a theoretical foundation for adopting adaptive deep learning models over traditional static forecasting methods.

Researchers have increasingly introduced machine learning-based models to overcome these limitations. Various supervised learning algorithms, particularly tree-based and margin-based models including Random Forest (RF), Support Vector Machine (SVM), and Gradient Boosting Tree (GBT), have found wide application in supply chain forecasting tasks [8, 18]. These models offer flexibility in capturing complex nonlinear structures, yet they inherently lack mechanisms to reflect the temporal dependencies of time-series data. Moreover, they often require substantial efforts in feature engineering to yield satisfactory results [19, 20].

More recently, deep learning-based models have gained prominence for their ability to improve the precision of time-series forecasting. Long Short-Term Memory, particularly, addresses the vanishing gradient problem inherent in traditional Recurrent Neural Networks (RNNs) and demonstrates superior performance in modelling long-term dependencies, including in supply chain contexts [9, 21]. Bidirectional LSTM extends this capability by learning temporal sequences in forward and backward directions, capturing richer contextual information, which is particularly valuable in scenarios where inter-temporal interactions are crucial [10, 22]. Nonetheless, BiLSTM suffers from structural limitations such as information dilution in long sequences and insufficient integration of exogenous variables [15].

Researchers have proposed the Nonlinear Autoregressive with Exogenous Inputs (NARX) model to address the challenge of incorporating external factors. NARX leverages both lagged dependent variables and exogenous inputs to model nonlinear regression, making it effective in capturing influences from weather, promotions, or macroeconomic indicators-elements commonly observed in supply chain dynamics [11, 18, 23]. Aldahmani et al. [12] proposed a heterogeneous neural architecture combining BiLSTM and NARX, referred to as the Uni-Regression Deep Approximate Forecasting model, and empirically demonstrated its superior predictive performance over single-model approaches. However, the UDAF model employs a fixedweight mechanism (e.g., 0.4:0.3:0.3) to integrate outputs from each module, which fails to dynamically reflect the relative predictive capabilities that may vary over time. Parallel to these developments, advanced architectures such as DeepAR [24] and N-BEATS [25] have been established as key benchmarks, offering probabilistic forecasting capabilities and interpretable basis expansion, respectively.

To resolve this limitation, Smyl [13] introduced the concept of Dynamic Weight Fusion (DWF), a technique that adjusts weights dynamically based on model-specific prediction errors at each time step [26]. This method aims to enhance forecasting accuracy through real-time optimization of model contributions. Subsequent studies by Zhang et al. [14] validated the effectiveness of DWF by demonstrating its utility in frequency-based time-series forecasting across various scenarios.

Meanwhile, attention mechanisms have emerged as powerful tools to mitigate the information dilution problem in BiLSTM by emphasizing temporally significant points within sequences. The multi-head attention mechanism, as introduced in the Transformer architecture by Vaswani et al. [10], effectively captures relative importance across time steps and enhances forecasting accuracy by assigning greater weights to more informative positions. Lim et al. [15] incorporated this mechanism into the Temporal Fusion Transformer, enabling integrated processing of exogenous and sequential features. More recently, Nie et al. [27] introduced patch-based processing to enhance long-term dependency learning. However, subsequent studies such as Zeng et al. [28] have argued that simple linear models can often outperform complex Transformer-based architectures in time-series forecasting, highlighting the risks of structural overfitting. In addition, attention-based models require intricate preprocessing, including input normalization, positional embedding, and stabilization techniques, and may otherwise suffer from overfitting or performance instability [29, 30].

In summary, prior studies have demonstrated the complementary strengths of various approaches: statistical models for simplicity, machine learning for nonlinear pattern recognition, deep learning for long-sequence modelling, NARX for exogenous variable incorporation, and BiLSTM for contextual learning. However, integrated architectures that concurrently support dynamic weight adaptation and emphasize temporally significant moments remain rare. Consequently, this study introduces three composite neural architectures, namely DWF_UDAF, MHA_UDAF, and DWFMHA_UDAF, that combine the BiLSTM and NARX frameworks with dynamic weight fusion and multi-head attention. These models are designed to meet the complex requirements of supply chain demand forecasting, including nonlinearity, long-term dependency, external factor integration, temporal salience, and dynamic contribution adjustment, and are empirically validated through comparative experiments.

3 METHODOLOGY

This study proposes an enhanced composite neural architecture for demand forecasting that integrates a Nonlinear Autoregressive model with exogenous inputs (NARX) and a Bidirectional Long Short-Term Memory (BiLSTM) network, augmented by two additional components, specifically Dynamic-Weight Fusion and multi-head attention. A key novelty of this study lies in the implementation of a gradient-based Adaptive Weight Optimizer. Instead of relying on complex reinforcement learning agents, this module utilizes a direct gradient-descent mechanism to perform online updates of fusion weights, enabling efficient adaptation to demand volatility.

In contrast to previous DWF studies using static or heuristically adapted weights, this approach enables online learning of time-varying weight distributions tailored to heterogeneous forecasters like BiLSTM and NARX. The aim is to overcome the limitations of fixed-weight ensemble structures while improving the handling of long-term dependencies and the incorporation of exogenous variables. Fig. 1 provides a visual representation of the overall BiLSTM-NARX composite forecasting model.

In the first stage illustrated in Fig. 1, three prediction modules, namely NARX, BiLSTM, and a general recurrent neural network (GeneralRNN), process the input time series independently to generate individual forecasts. The NARX module is designed to capture nonlinear effects from exogenous variables, such as promotions or external shocks. The BiLSTM module, with its bidirectional architecture, effectively learns long-term dependencies within the sequence. The GeneralRNN serves as a baseline model for comparative purposes, representing a standard sequence-processing architecture.

In the second stage, the outputs of the three modules are combined using fixed weights in a structure known as Uni-Regression Deep Approximate Forecasting (UDAF). In this study, the weights were set to 0.4 for NARX, 0.3 for BiLSTM, and 0.3 for RNN, reflecting experimentally optimized ratios derived from prior research [11]. However, this architecture relies on predefined, non-learnable weights, which inherently limit its ability to adapt to temporal variations in model-specific predictive accuracy.

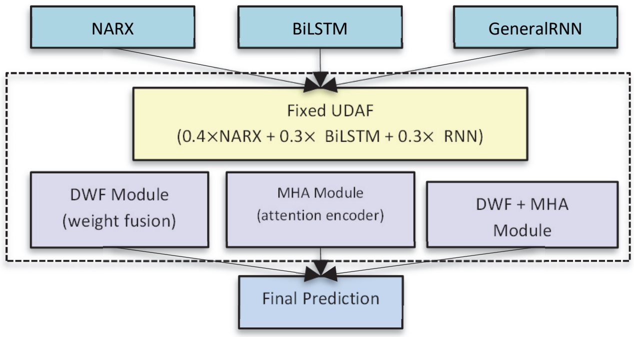

Figure 1 Architecture of the proposed BiLSTM-NARX composite model with dynamic weight function and multi-head attention

To address this limitation, the third stage introduces three enhanced fusion strategies. The first is the Dynamic Weight Fusion (DWF) module, which dynamically adjusts the weights assigned to the BiLSTM and NARX outputs based on their prediction performance at each time step. The second is the Multi-Head Attention (MHA) module, which is applied in the hidden states of the BiLSTM to redistribute attention across time steps by quantifying their relative importance. This mechanism alleviates the issue of information dilution in long sequences. The third and final structure is a combined DWF_MHA module, which integrates both dynamic weight adaptation and temporal emphasis to maximize predictive performance.

Each structure operates on the same input data but utilizes different integration mechanisms to generate final predictions. Through comparative experiments, the study evaluates the performance of these advanced architectures against the baseline UDAF model, aiming to identify an optimal fusion strategy capable of overcoming the inherent constraints of fixed-weight ensembles.

3.1 BiLSTM Module

While maintaining the long-sequence modelling capability of conventional LSTM networks, the BiLSTM module is designed to capture bidirectional long-term dependencies in input time-series data by employing a bidirectional recurrent structure. This study adopts the standard BiLSTM architecture proposed by Graves et al. [21] and Hochreiter & Schmidhuber [9]. Since the gating mechanisms and cell state updates follow the conventional formulation found in standard literature, detailed mathematical descriptions are omitted for brevity.

The BiLSTM module was trained using the Adam optimization with a learning rate of 0.001. The model was trained for 100 epochs, and early stopping was applied when the validation loss did not decrease for 10 consecutive epochs. The length of the input sequence was set to 7 days based on a sliding-window approach, so that the model could capture the weekly seasonality inherent in store-level demand.

Unlike prior approaches that applied static averaging or fixed weights to aggregate hidden states, this study incorporates Multi-Head Attention and Dynamic Weight Fusion modules to enable both temporal relevance weighting and context-sensitive output refinement. This not only alleviates the issue of information dilution in long sequences but also contributes significantly to enhancing forecasting performance in dynamic supply chain environments.

Previous studies typically aggregated the hidden state outputs of BiLSTM using simple averaging or fixed weights, which failed to capture the varying importance of information across time steps. This paper integrates an attention mechanism into the BiLSTM output to emphasize critical time points, thereby mitigating the issue of information dilution in long sequences and contributing to improved prediction accuracy.

3.2 NARX Module

The second key forecasting module employed in this study is the Nonlinear Autoregressive model with exogenous inputs. As a nonlinear autoregressive model, NARX can incorporate both historical values of the dependent variable and lagged values of exogenous variables, thereby effectively capturing the influence of complex external factors. In supply chain demand forecasting, where various external variables such as price, promotions, weather, and economic indicators can impact demand, the ability of the NARX model to explicitly include these factors as input features provides a significant structural advantage.

The NARX model is employed to incorporate both historical demand values and lagged exogenous variables. This study implements the NARX structure using a multi-layer perceptron (MLP) as described in standard time-series forecasting literature [11, 24]. The model takes the lagged dependent variables and exogenous inputs as a concatenated vector to predict future demand. Model training is conducted using backpropagation with the Adam optimizer, and the loss function is defined as the mean absolute error (MAE).

Whereas traditional forecasting models often handle exogenous variables using single-point values or simple averages, the NARX model integrates temporally lagged exogenous inputs. This enables more precise modelling of time-varying external shocks and their effects on demand. In this study, the NARX module operates in parallel with the BiLSTM module, and their respective outputs are later combined through the Dynamic Weight Fusion module and the multi-head attention module to generate the final forecast.

3.3 Dynamic Weight Function Module

Conventional ensemble-based forecasting models often rely on static-weight combinations when aggregating outputs from different predictive modules. For example, in the Uni-Regression Deep Approximate Forecasting model combining BiLSTM and NARX, the forecast is obtained by linearly blending the outputs of three modules with predefined weights (e.g., 0.4, 0.3, 0.3). However, such static-weight schemes fail to capture the time-varying performance of each module, ignoring the fact that the relative predictive accuracy of individual modules may vary significantly across time steps.

To address this limitation, this study adopts a Dynamic Weight Fusion approach that adaptively adjusts the contribution of each predictive module at every time step based on its real-time forecasting performance. Specifically, the outputs of the BiLSTM and NARX modules are fused using dynamically learned weights optimized to minimize overall forecasting error.

Let $\hat{y}_t^{\mathrm{BiLSTM}}$ and $\hat{y}_t^{\mathrm{NARX}}$ denote the forecasted values from BiLSTM and NARX at time t, respectively. Then, the DWF-based final prediction $\hat{y}_t^{\mathrm{DWF}}$ is defined as Eq. (1).

$$w_1 \cdot \hat{y}_t^{\mathrm{BiLSTM}} + w_2 \cdot \hat{y}_t^{\mathrm{NARX}}, \text{ where } w_1 + w_2 = 1,\ w_1,\ w_2 \geq 0 \tag{1}$$

The objective function for learning the optimal weights over the entire forecasting period t = 1, …, T is given by Eq. (2).

$$\min_{w_1, w_2} \sum_{t=1}^{T} \left| y_t - \left(w_1 \cdot \hat{y}_t^{\mathrm{BiLSTM}} + w_2 \cdot \hat{y}_t^{\mathrm{NARX}} \right) \right| \tag{2}$$

Here, y_t denotes the actual demand value at time t, and the loss function is described as the Mean Absolute Error. Weights w_1 and w_2 are treated as learnable parameters, randomly initialized and updated using the Adam optimizer.

Compared to static fusion, the DWF module provides a more flexible structure that dynamically emphasizes the stronger module at each time step. For instance, when contextual dependencies dominate, the model increases w_1 to prioritize BiLSTM; conversely, when exogenous factors prevail, w_2 becomes dominant, enhancing the influence of NARX. This adaptivity allows the model to respond more sensitively to structural changes in the time-series data and stabilizes predictive performance.

To ensure the constraint, $w_1 + w_2 = 1$, normalization is applied, either through softmax or simple normalization. This study adopts a normalized weighted sum approach, maintaining interpretability by keeping the weights within [0, 1] with their sum fixed at 1.

-Input: BiLSTM Forecasts ($\hat{y}_t^{\mathrm{BiLSTM}}$), NARX Forecasts ($\hat{y}_t^{\mathrm{NARX}}$),
Actual Demand (y_t)
-Output: Final Hybrid Forecast (Y_f), Optimized Weights (w_1, w_2)
-Parameters: Learning rate α, Epochs E
1: Initialize weights w_1, w_2 randomly
2: For epoch = 1 to E do:
3: Compute combined forecast: $\hat{y}_t = w_1 \cdot \hat{y}_t^{\mathrm{BiLSTM}} + w_2 \cdot \hat{y}_t^{\mathrm{NARX}}$
4: Calculate Loss (MAE): $L = |y_t - \hat{y}_t|$
5: Update weights using Adam Optimizer based on gradient of L
6: Apply constraints (Normalization):
$w_1 \leftarrow \dfrac{w_1}{w_1 + w_2}$, $w_2 \leftarrow \dfrac{w_2}{w_1 + w_2}$
7: End For
8: Return $Y_f = w_1 \cdot \hat{y}_t^{\mathrm{BiLSTM}} + w_2 \cdot \hat{y}_t^{\mathrm{NARX}}$

Figure 2 Algorithm: dynamic weight function (DWF) optimization

The DWF module fuses time-specific predictions from BiLSTM and NARX in a way that adapts sensitively to temporal variations. It thus contributes a structurally robust forecasting framework that achieves high accuracy and generalizability in supply chain demand forecasting tasks. The iterative process for optimizing the dynamic weights and minimizing the forecasting error is formally described in the algorithm below (Fig. 2).

3.4 Multi-Head Attention Module

While BiLSTM captures long-term dependencies in time-series data, it suffers from information dilution, particularly for intermediate time steps as the sequence length increases. To mitigate this issue, this study incorporates a Multi-Head Attention mechanism on top of the hidden states produced by the BiLSTM module. The attention mechanism learns the relative significance of time steps in the sequence, enabling the model to emphasize the pertinent temporal features for forecasting.

The attention mechanism, proposed by Vaswani et al. [10] as a fundamental element of the Transformer architecture, has proven effective in time-series forecasting applications, notably in the Temporal Fusion Transformer [15]. In this study, the hidden-state sequence of the BiLSTM serves as the input to the attention module, which adjusts the attention weights across time steps to generate an attention-weighted representation.

Given an input sequence $X \in R^{T \times d}$, the Query, Key, and Value matrices are calculated as in Eq. (3).

$$Q = XW^Q, K = XW^K, V = XW^V \tag{3}$$

Here, W^Q, W^K, $W^V \in R^{d \cdot d^k}$ are learnable weight matrices, where d^k indicates the dimensionality of each attention head. The attention weights are calculated by applying a scaled dot-product between the Query and Key matrices, followed by a softmax normalization. The resulting attention output is defined in Eq. (4).

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{4}$$

The multi-head attention mechanism performs this operation in parallel across h independent attention heads. The outputs from each head are then concatenated and projected via a linear transformation to create the final output, as defined in Eq. (5).

$$\begin{gathered}\text{MHA}(X) = \\ \text{Concat}(\text{head}_1, \text{head}_2, ..., \text{head}_h)W^O, \\ W^O \in R^{Hd_k \cdot d}\end{gathered} \tag{5}$$

Each attention head operates independently in a distinct subspace, which permits the model to obtain various positional and semantic dependencies across the sequence. The number of heads was set to $h = 4$, and the dimension of each attention subspace was set to $d_k = 32$. To align the dimensionality of the final output with that of the BiLSTM hidden states, the output projection matrix $W^O \in R^{Hd_k \cdot d}$ was applied.

This multi-head attention module was applied to the BiLSTM hidden state sequence, $H = \{h_1, h_2, …, h_T\}$, generating the attention-weighted representation H'. By emphasizing important temporal steps within the sequence, this mechanism ensures that salient information contributing to prediction is not diluted. Particularly in time-series forecasting, where specific time points (e.g., promotional start dates or holidays) may exert significant influence on demand, the attention scores dynamically increase at those steps, allowing the model to prioritize such critical inputs.

During training, softmax normalization was applied to the attention weights, and dropout regularization was applied to mitigate overfitting. Although the introduction of MHA increases computational complexity compared to the BiLSTM-only structure, it offers practical benefits in preserving crucial information and improving prediction accuracy. Moreover, by analyzing the resulting attention scores, this study enhances model interpretability, providing insights into which time points were most influential in the forecasting process.

3.5 Integrated Architecture and Training Procedure

A layered ensemble forecasting structure is introduced in this study, combining statistical analysis, machine learning methods, and recurrent neural models to improve prediction accuracy [31]. All experiments were conducted in the Google Colab environment using Python. This study utilizes the Rossmann Store Sales time-series dataset provided by Kaggle. Specifically, the analysis focuses on Store 1, which serves as a methodological testbed due to its exceptionally high sales volatility and sensitivity to diverse exogenous factors (e.g., promotions and holidays). Validating the model on this complex time-series allows for a rigorous assessment of the Dynamic Weight Fusion (DWF) mechanism's adaptability, rather than a broad generalization across all store types. After excluding dates when Open = 0, a total of 799 daily observations remained. Of these, the first 757 records, ordered chronologically, are used for training and validation, while the last 42 records (approximately six weeks) serve as the hold-out test set for all experiments. The dataset was pre-processed to construct a time-series input matrix including date-related features, which were then utilized for model training and testing. This study addresses missing value issues as follows. Competition Distance was imputed using the median value; Promo2SinceWeek and Promo2SinceYear were set to zero; and PromoInterval was replaced with an empty string to indicate the absence of promotion intervals. All other variables are complete. The preprocessing code, including exact filtering steps and package versions, is available in the accompanying Colab notebook to ensure full reproducibility. StateHoliday was converted using the same 0-3 mapping described above, and retail_feature was calculated exactly as in the accompanying notebook by first taking a seven-day rolling mean of Sales and then applying the competition-distance interaction defined earlier.

The proposed forecasting framework is composed of three primary stages: (1) independent training and prediction using individual modules, (2) fixed- and dynamic-weight forecast fusion (UDAF, DWF_UDAF), and (3) attention-based post-processing modules (MHA_UDAF, DWFMHA_UDAF).

The neural network modules, NARX, BiLSTM, and GeneralRNN, were implemented using the PyTorch framework. All models were trained with a fixed sequence length of 7 to align with the intrinsic weekly seasonality of retail sales data. For ARIMAX, the model was trained using the pmdarima.auto_arima function, which internally optimized the (p, d, q) parameters. The exogenous variables used included Open, Promo, StateHoliday, CompetitionDistance, and retail_feature. For Random Forest and XGBoost, a fixed hyperparameter set was employed with 100 estimators and a random seed of 42, without further grid search. While this limited tuning may not represent optimal configurations, it ensures consistency across all models for a fair baseline comparison. The number of hidden units was fixed at 64 for all recurrent models, including NARX, BiLSTM, and GeneralRNN. For the models employing attention mechanisms (DWFMHA_UDAF and MHA_UDAF), a dropout rate of 0.1 and an embedding dimension of 16 were selected to prevent overfitting given the limited sample size. The number of attention heads was set to 4, balancing the need for diverse temporal feature extraction with computational efficiency. These configurations were chosen to balance model capacity and overfitting risk given the limited sample size (approximately 800 training samples) and the observed tendency of attention heads to collapse when using larger dimensions or more heads. The optimizer for all deep learning models, including the NARX MLP, was Adam. A fixed random seed was used to ensure reproducibility. The loss function was set to mean squared error (MSE), and optimization was performed using the Adam algorithm. Input sequences were generated with a 7-day sliding window and the following six features: Sales, Open, Promo, StateHoliday (ordinal encoding: 0 = none, 1 = '*a*', 2 = '*b*', 3 = '*c*'), CompetitionDistance, and retail_feature $\log\left(1+\overline{\mathrm{Sales}_7}\cdot\left(1+\frac{1}{1+\mathrm{CompetitionDistance}}\right)\right)$. Here, $\overline{\mathrm{Sales}_7}$ is the 7-day moving average of Sales. After training, each module generated individual predictions on the test set, which were then used in subsequent fusion strategies.

The fixed weight ensemble model, UDAF, performed a linear combination of the outputs from NARX (0.4), BiLSTM (0.3), and GeneralRNN (0.3), based on predefined weights.

Next, the DWF_UDAF model introduced dynamic weights learned through a gradient-based Multi-Layer Perceptron (MLP) module, referred to as the WeightOptimizer. This module dynamically adjusted weights based on the real-time predictions of the three base models. The predicted weights were normalized using a softmax function, and the forecast was calculated as a weighted sum of three outputs.

The final stage involves the integration of attention mechanisms to enhance feature representation. In the MHA_UDAF model, the Multi-Head Attention mechanism is applied directly to the hidden-state sequence of the BiLSTM module, allowing the model to weigh critical time steps before generating the final prediction. Similarly, in DWFMHA_UDAF, these attention-weighted BiLSTM features are combined with the NARX output, utilizing the DWF mechanism to dynamically optimize the fusion weights.

In the DWFMHA_UDAF architecture, the Multi-Head Attention module processes the hidden states of the BiLSTM to extract context-aware features. These attention-enhanced features are then fused with the NARX output via the Dynamic Weight Fusion mechanism, ensuring that temporal importance is weighed before the final integration. This module was implemented as a PyTorch class.

The forecasting performance of ten models, including ARIMAX, Random Forest, XGBoost, NARX, BiLSTM, GeneralRNN, UDAF, DWF_UDAF, MHA_UDAF, and DWFMHA_UDAF, was compared. Model performance was assessed using standard quantitative metrics, specifically Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the coefficient of determination (R^2). All predictions were flattened into one-dimensional arrays and aligned to match the same forecast horizon. To evaluate the statistical significance of performance differences across models, One-way ANOVA with Tukey's HSD post hoc test was conducted, as detailed in Section 4.

4 RESULTS

This study compared the predictive performance of ten forecasting models, including ARIMAX, Random Forest, XGBoost, NARX, BiLSTM, GeneralRNN, UDAF, DWF_UDAF, DWFMHA_UDAF, and MHA_UDAF, under identical experimental conditions. To enhance reliability and ensure temporal independence, the evaluation was conducted over 20 independent trials using randomly selected continuous time-blocks. In each trial, strictly chronological splitting was applied (training on past data, testing on future data) to preserve causality and prevent data leakage. Performance variability was reported as the standard deviation, and significance was assessed via ANOVA and Tukey's HSD. This procedure ensured that no data overlapped between the training and test sets during any run, thereby maintaining independence across evaluations. To validate the statistical significance of the results, this study employed One-way ANOVA and Tukey's HSD test. While non-parametric tests (e.g., Friedman test) are often used for ranking models across multiple datasets, ANOVA is appropriate in this study as we compare the distribution of errors derived from 20 independent Monte Carlo cross-validation trials on a single focused dataset. The sample size (N = 20 per model) is sufficient to satisfy the robustness assumptions of parametric testing, allowing for a more sensitive analysis of mean performance differences. The models encompassed statistical, machine learning, deep learning, and ensemble-based neural-network architectures, and the experiments used the Rossmann Store Sales time-series dataset provided by Kaggle.

Model performance was evaluated using three standard quantitative metrics commonly applied in time-series regression: mean absolute error (MAE), root mean square error (RMSE), and the coefficient of determination (R^2). Tab. 1 summarizes the mean MAE ± standard deviation of the model on 20 independent runs.

Among the models evaluated, DWF_UDAF consistently recorded the highest level of predictive accuracy, recording an MAE of 0.147 ± 0.005, an RMSE of 0.214, and an R^2 of 0.936. This result suggests that the DWF approach, which dynamically adjusts the weights based on time-specific prediction errors, effectively overcomes the limitations of the fixed weight fusion method used in UDAF.

Table 1 Summary of average predictive performance by model

Model	MAE(mean ± SD)	RMSE	R^2
DWF_UDAF	0.147 ± 0.005	0.214	0.936
ARIMAX	0.164 ± 0.006	0.206	0.938
UDAF	0.177 ± 0.006	0.245	0.916
GeneralRNN	0.187 ± 0.015	0.254	0.910
BiLSTM	0.187 ± 0.016	0.251	0.912
NARX	0.191 ± 0.013	0.261	0.905
RF	0.195 ± 0.008	0.258	0.903
XGB	0.207 ± 0.011	0.283	0.883
DWFMHA_UDAF	0.607 ± 0.085	0.843	−0.012
MHA_UDAF	0.670 ± 0.112	0.914	−0.182

In contrast, the fixed-weight UDAF model demonstrated stable performance by MAE of 0.177 ± 0.006, an RMSE of 0.245, and an R^2 of 0.916; however, it was less accurate compared to DWF_UDAF. The traditional statistical model ARIMAX attained the highest R^2 score of 0.938, with an MAE of 0.164 ± 0.006 and RMSE of 0.206, but it failed to capture nonlinearities and complex interactions with exogenous variables adequately.

Recurrent neural network models such as BiLSTM, NARX, and GeneralRNN showed similar performance, with MAE values of 0.187 ± 0.016, 0.191 ± 0.013, and 0.187 ± 0.015, respectively, indicating structural limitations in capturing long-term dependencies or incorporating exogenous variables effectively. Machine learning models such as RF and XGB yielded MAE values of 0.195 ± 0.008 and 0.207 ± 0.011, respectively, suggesting that their inability to fully account for time-series characteristics led to decreased performance.

The models incorporating the MHA module, DWFMHA_UDAF and MHA_UDAF, exhibited significantly lower performance, with MAE values of 0.607 ± 0.085 and 0.670 ± 0.112, and RMSE values of 0.843 and 0.914. The findings suggest that attention-driven architectures may only function effectively when accompanied by carefully engineered input preprocessing techniques.

To evaluate whether the performance differences among models were statistically significant, a one-way ANOVA test was conducted based on MAE values. The results showed an F-value of 366.44 and a p-value of 5.56×10^{-115}, both below the significance level of 0.05, indicating that the differences in mean MAE models were statistically significant (Tab. 2).

Table 2 ANOVA results on MAE values across models

Source	Sum Sq	df	Mean Sq	F-value	p-value
Model	6.759	9	0.751	366.44	5.56×10^{-115}
Residual	0.389	190	0.002		

To further verify the statistical significance of the experimental results, Tukey's Honest Significant Difference (HSD) posthoc test is performed, and the results are summarized in Tab. 3.

The DWF_UDAF model reveals statistically significant differences ($p < 0.001$) when in contrast with DWFMHA_UDAF and MHA_UDAF. This demonstrates that the proposed DWF-based structure is statistically superior to both attention-based and fixed weight-based models.

In particular, the MAE difference between DWF_UDAF and DWFMHA_UDAF was −0.4597, and −0.5234 when compared with MHA_UDAF. Both differences were significant at the 95% confidence level. However, no statistically significant differences were declared when comparing DWF_UDAF with UDAF, GeneralRNN, or BiLSTM, although DWF_UDAF consistently achieved a lower average MAE than these models.

In pairwise comparisons with other baseline models, namely ARIMAX, RF, XGB, BiLSTM, GeneralRNN, UDAF, and NARX, no statistically significant differences were observed at the 5% significance level, except for the RF and XGB models. The comparisons with RF (p = 0.0306) and XGB (p = 0.0016) showed significant differences, suggesting that DWF_UDAF outperformed traditional regression-based or ensemble-based models in predictive accuracy.

DWF_UDAF consistently achieves lower MAE values compared to recurrent neural network models like BiLSTM and GeneralRNN, indicating its robustness and stability of performance across different experimental runs, even though no statistically significant differences were found between the models.

Table 3 Tukey's HSD post-hoc test results between prediction models

Group1	Group2	Mean diff	p-adj	lower	upper	reject
ARIMAX	BiLSTM	0.0232	0.835	−0.0226	0.0691	FALSE
	DWFMHA_UDAF	0.4424	0	0.3965	0.4882	TRUE
	DWF_UDAF	−0.0174	0.9696	−0.0632	0.0285	FALSE
	GeneralRNN	0.0227	0.8548	−0.0232	0.0685	FALSE
	MHA_UDAF	0.506	0	0.4602	0.5518	TRUE
	NARX	0.0271	0.6721	−0.0187	0.073	FALSE
	RF	0.0308	0.4921	−0.015	0.0767	FALSE
	UDAF	0.0123	0.9975	−0.0336	0.0581	FALSE
	XGB	0.0428	0.0896	−0.003	0.0886	FALSE
BiLSTM	DWFMHA_UDAF	0.4191	0	0.3733	0.4649	TRUE
	DWF_UDAF	−0.0406	0.1319	−0.0864	0.0052	FALSE
	GeneralRNN	−0.0006	1	−0.0464	0.0453	FALSE
	MHA_UDAF	0.4828	0	0.4369	0.5286	TRUE

Table 3 Tukey's HSD post-hoc test results between prediction models - continuation

Group1	Group2	Mean diff	*p*-adj	lower	upper	reject
	NARX	0.0039	1	−0.0419	0.0497	FALSE
	RF	0.0076	0.9999	−0.0382	0.0534	FALSE
	UDAF	−0.011	0.9989	−0.0568	0.0349	FALSE
	XGB	0.0196	0.936	−0.0263	0.0654	FALSE
DWFMHA_UDAF	DWF_UDAF	−0.4597	0	−0.5056	−0.4139	TRUE
	GeneralRNN	−0.4197	0	−0.4655	−0.3739	TRUE
	MHA_UDAF	0.0636	0.0006	0.0178	0.1095	TRUE
	NARX	−0.4152	0	−0.4611	−0.3694	TRUE
	RF	−0.4115	0	−0.4574	−0.3657	TRUE
	UDAF	−0.4301	0	−0.4759	−0.3843	TRUE
	XGB	−0.3996	0	−0.4454	−0.3537	TRUE
DWF_UDAF	GeneralRNN	0.04	0.1454	−0.0058	0.0859	FALSE
	MHA_UDAF	0.5234	0	0.4775	0.5692	TRUE
	NARX	0.0445	0.0651	−0.0013	0.0903	FALSE
	RF	0.0482	0.0306	0.0024	0.094	TRUE
	UDAF	0.0296	0.5512	−0.0162	0.0755	FALSE
	XGB	0.0602	0.0016	0.0143	0.106	TRUE
GeneralRNN	MHA_UDAF	0.4833	0	0.4375	0.5292	TRUE
	NARX	0.0045	1	−0.0414	0.0503	FALSE
	RF	0.0082	0.9999	−0.0377	0.054	FALSE
	UDAF	−0.0104	0.9993	−0.0562	0.0354	FALSE
	XGB	0.0201	0.9239	−0.0257	0.066	FALSE
MHA_UDAF	NARX	−0.4789	0	−0.5247	−0.433	TRUE
	RF	−0.4752	0	−0.521	−0.4293	TRUE
	UDAF	−0.4937	0	−0.5396	−0.4479	TRUE
	XGB	−0.4632	0	−0.509	−0.4174	TRUE
NARX	RF	0.0037	1	−0.0421	0.0495	FALSE
	UDAF	−0.0149	0.9895	−0.0607	0.031	FALSE
	XGB	0.0157	0.9848	−0.0302	0.0615	FALSE
RF	UDAF	−0.0186	0.9533	−0.0644	0.0273	FALSE
	XGB	0.012	0.9979	−0.0339	0.0578	FALSE
UDAF	XGB	0.0305	0.5068	−0.0153	0.0764	FALSE

The analysis results empirically demonstrate that the proposed DWF-based BiLSTM-NARX neural ensemble architecture outperforms existing models in predictive accuracy and statistical significance. This finding supports the effectiveness of an adaptive fusion structure that dynamically responds to temporal variability and exogenous factors, thereby enabling substantial improvements in forecasting performance.

5 DISCUSSION

This study proposed novel neural ensemble forecasting architectures, namely DWF_UDAF, MHA_UDAF, and DWFMHA_UDAF, that integrate the BiLSTM-NARX model with the Dynamic Weight Fusion algorithm to enhance the accuracy of supply chain demand forecasting. It conducted a comprehensive comparative experiment involving a total of 10 forecasting models. The outcome demonstrated that the DWF_UDAF model obtained the predictive performance in terms of MAE, RMSE, and R^2, and statistical tests such as ANOVA and Tukey's HSD confirmed the significance of these differences. While the DWF_UDAF model showed slightly higher RMSE than the ARIMAX model, it demonstrated significantly lower MAE and higher R^2 across multiple runs, indicating more accurate and consistent predictions overall. Given the high sensitivity of RMSE to outliers, this study placed greater emphasis on MAE and R^2 for overall model ranking, which better reflect average error and explanatory power in practical demand forecasting scenarios.

Notably, the DWF_UDAF outperformed the conventional ensemble model UDAF, which uses fixed weight averaging, and also exhibited more stable and lower prediction errors compared to Transformer-based attention-fusion models such as MHA_UDAF and DWFMHA_UDAF. This study analyzes that the poor performance of MHA-based models stems from 'Attention Collapse' due to data sparsity. Since Transformer-based architectures generally require large-scale datasets to learn generalized temporal dependencies, applying them to a single time-series with limited samples (N =757) caused the attention mechanism to overfit to specific noise patterns rather than learning meaningful trends. The analysis of attention scores confirms this, revealing that weights were disproportionately concentrated on irrelevant time steps, failing to distribute attention effectively. A qualitative inspection of the learned weights revealed that higher weights were consistently assigned to the NARX module during promotional weeks and periods with external disruptions, indicating that the DWF mechanism effectively prioritized modules better suited to exogenous variability.

Furthermore, machine learning models such as XGB and RF showed inferior prediction accuracy compared to deep learning-based models, primarily due to their limitations in capturing dynamic time-series features and exogenous variables. It implies that deep learning models with adaptive structures are more suitable for complex forecasting problems like supply chain demand, where external factors play a critical role.

In summary, this study goes beyond simple model performance comparison. It empirically validates the superiority of DWF-based fusion models through a multidimensional evaluation of forecasting accuracy, statistical significance, and structural soundness. Further interpretation of diagnostic plots (not shown) indicated that prediction errors were more volatile during promotional weeks and around holidays, particularly in fixed-weight and attention-based models. This suggests that models lacking adaptive mechanisms fail to adjust effectively under exogenous shocks. From a managerial perspective, the proposed DWF-based approach enables earlier detection of demand surges. It stabilizes inventory planning, helping to reduce stockouts and mitigate the bullwhip effect across upstream suppliers. These results highlight the potential of the proposed method in real-world domains such as supply chain management, inventory optimization, and sales strategy formulation, suggesting that dynamic weight-based fusion strategies are a promising direction for future time-series forecasting frameworks. In this case study on the Rossmann Store Sales dataset, the gradient-based DWF module combined with a heterogeneous BiLSTM-NARX backbone achieved the best performance among the ten benchmark models considered.

6 CONCLUSION

The study was designed to strengthen supply chain demand forecasting by proposing three neural ensemble forecasting models, DWF_UDAF, MHA_UDAF, and DWFMHA_UDAF, which combine BiLSTM and NARX with Dynamic Weight Fusion and multi-head attention mechanisms. The models were evaluated through extensive comparative experiments.

The experimental results confirmed that the DWF_UDAF model achieved the best performance among all ten models, including statistical model (ARIMAX), machine learning models (RF, XGB), deep learning models (BiLSTM, NARX, GeneralRNN), neural-ensemble models (UDAF), and attention fusion models (MHA_UDAF, DWFMHA_UDAF). Although the ARIMAX model showed slightly lower RMSE in certain runs, the DWF_UDAF model consistently achieved better MAE and R^2 values, indicating more stable and accurate predictions across different conditions. Therefore, MAE and R^2 were prioritized as the primary evaluation metrics for determining overall performance. This superiority was also statistically validated through ANOVA and post hoc Tukey tests.

The implications of this study are the following:

1) The DWF method dynamically adjusts the weights based on the time-specific performance of forecasting models, allowing it to effectively handle volatile demand patterns.

2) Although MHA-integrated architectures have theoretical advantages, their excessive complexity may degrade forecasting performance, indicating that attention-based mechanisms are not universally suitable for all problem domains.

3) In scenarios like supply chain forecasting, where time-series and exogenous variables interact, neural ensemble architectures based on dynamic weighting have been empirically validated as one of the most effective approaches.

Future research can extend this study in several directions. First, comparative studies against broader adaptive architectures, such as Mixture of Experts (MoE) and Temporal Mixture Models, should be conducted to further validate the relative efficacy of the proposed heterogeneous framework. Additionally, novel attention fusion architectures that combine the DWF mechanism with Transformer-based models should be evaluated. Second, the generalizability of the proposed models to diverse industrial datasets (e.g., pharmaceutical, retail, and smart factory) needs to be assessed. Third, the development of lightweight models suitable for real-time deployment should be pursued in parallel.

In conclusion, this study empirically demonstrated that dynamic and adaptive weight fusion strategies outperform conventional static prediction structures in terms of practical, statistical, and structural performance in supply chain demand forecasting. It contributes both practical and academic value by providing design guidelines for future time-series forecasting algorithms. A primary limitation of this study is the empirical validation on a single, albeit complex, dataset (Store 1). While this sufficiently demonstrates the efficacy of the adaptive DWF methodology, future research should extend this framework to diverse store types (e.g., Models A, B, C, D) and other industrial domains to fully establish its generalizability. In practical terms, the DWF-based approach can assist supply chain managers in reducing inventory buffers and mitigating bullwhip effects by enabling more responsive and stable demand forecasts.

Acknowledgement

This study was done by research funds from Gwangju University in 2026.

7 REFERENCES

[1] Syntetos,A. A. & Boylan, J. E. (2005). The accuracy of intermittent demand estimates. *International Journal of Forecasting*, *21*(2), 303-314. https://doi.org/10.1016/j.ijforecast.2004.10.001

[2] Pajic, V., Andrejic, M., & Kilibarda, M. (2025). Procurement optimization by selecting efficient suppliers using DEA-FUCOM-COCOSO approach and solving order allocation problem. *Facta Universitatis, Series: Mechanical Engineering, 23*(1), 109–125. https://doi.org/10.22190/FUME220210031P

[3] Noor, Z. Z. (2025). Sustainable production management in circular economy supply chains. *International Journal of Industrial Engineering and Management, 16*(2), 204–215. https://doi.org/10.24867/IJIEM-384

[4] Zhang, H., Guo, Y. W., Hou, Y., Tang, L., & Deveci, M. (2024). Improved whale optimization algorithm for supply chain financial risk assessment of cloud warehouse platform. *Advances in Production Engineering & Management, 19*(3), 395–407. https://doi.org/10.14743/apem2024.3.515

[5] Ivanov, D. (2020). Predicting the impacts of epidemic outbreaks on global supply chains: A simulation-based analysis on the COVID-19 pandemic. *Transportation Research Part E: Logistics and Transportation Review*, *136*, 101922. https://doi.org/10.1016/j.tre.2020.101922

[6] Wieland, A. (2021). Dancing the supply chain: Toward transformative supply chain management. *Journal of Supply Chain Management*, *57*(1), 58-73. https://doi.org/10.1111/jscm.12248

[7] Hyndman, R. J. & Athanasopoulos, G. (2021). *Forecasting: Principles and Practice, 3rd edition*. OTexts, Melbourne, Australia.

[8] Feizabadi, J. (2022). Machine learning demand forecasting and supply chain performance. *International Journal of Logistics Research and Applications*, *25*, 119-142.

https://doi.org/10.1080/13675567.2020.1803246

[9] Hochreiter, S. & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

[10] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J.,Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, L.(2017). Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 1-15.

[11] Box, G. E., Jenkins, G. M., & Reinsel, G. C. (2015). *Time Series Analysis: Forecasting and Control*. John Wiley & Sons, New Jersey, USA. https://doi.org/10.1111/jtsa.12194

[12] Aldahmani, E., Alzubi, A., & Iyiola, K. (2024). Demand Forecasting in Supply Chain Using Uni-Regression Deep Approximate Forecasting Model. *Applied Sciences*, *14*, 8110. https://doi.org/10.3390/app14188110

[13] Smyl, S. (2020). A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. *International Journal of Forecasting*, *36*(1), 75-85. https://doi.org/10.1016/j.ijforecast.2019.05.012

[14] Zhang, X., Zhao, S., Song, Z., Guo, H., Zhang, J., & Zheng, C. (2024). Towards a dynamic fusion of frequencies in time-series forecasting, *MM2024*, 4729-4737. https://doi.org/10.1145/3664647.3681210

[15] Lim, B., Arik, S. O., Loeff, N., & Pfister, T. (2021). Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, *37*(4), 1740-1760. https://doi.org/10.1016/j.ijforecast.2021.03.012

[16] Jiang, Y., Feng, T., & Huang, Y. (2024). Antecedent configurations toward supply chain resilience: The joint impact of supply chain integration and big data analytics capability. *Journal of Operations Management*, *70*(2), 257-284. https://doi.org/10.1002/joom.1282

[17] Chen, K., Li, Y., & Linderman, K. (2022). Supply network resilience learning: An exploratory data analytics study. *Decision Sciences*, *53*(1), 8-27. https://doi.org/10.1111/deci.12513

[18] Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2018). A comparison of ARIMA and LSTM in forecasting time series, *Proceedings of the IEEE International Conference on Machine Learning and Applications*, 1394-1401. https://doi.org/10.1109/ICMLA.2018.00227

[19] Cui, H., Wang, Y., & Feng, L. (2020). A deep learning approach for non-linear autoregressive time series prediction. *Pattern Recognition Letters*, *133*, 153-159. https://doi.org/10.1016/j.patrec.2020.03.014

[20] Maududy, R. & Nurdin, A. M. (2024). An architecture framework for supply chain management systems integrated with supervisory control and data acquisition functionality. *Journal of Logistics, Informatics and Service Science*, *11*(5), 38-51. https://doi.org/10.33168/JLISS.2024.0503

[21] Graves, A., Mohamed, A., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 6645-6649. https://doi.org/10.1109/ICASSP.2013.6638947

[22] Lin, J., Rivest, L. P., & Asselin, É. (2006). Nonlinear time series modeling with autoregressive neural networks. *Computational Statistics & Data Analysis*, *50*(11), 2965-2985. https://doi.org/10.1016/j.csda.2005.03.011

[23] Hu, H., Li, Y., & Li, M. (2024), Dual-channel supply chain coordination considering the consumer's perception of quality. *Economic Computation and Economic Cybernetics Studies and Research*, *58*(2), 198-216. https://doi.org/10.24818/18423264/58.2.24.12

[24] Salinas, D., Flunkert, V., Gasthaus, J., & Januschowski, T. (2020). DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, *36*(3), 1181-1191. https://doi.org/10.1016/j.ijforecast.2019.07.001

[25] Oreshkin, B. N., Carpov, D., Chapados, N., & Bengio, Y. (2020). N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. *Proceedings of the International Conference on Learning Representations (ICLR 2020)*.

[26] Chen, C. T., Ova, A., & Hung, W. Z. (2024). An MCDM method with dynamic weights for investment project selection. *Economic Computation and Economic Cybernetics Studies and Research*, *58*(2), 116-131. https://doi.org/10.24818/18423264/58.2.24.07

[27] Nie, Y., Nguyen, N. H., Sinthong, P., & Kalagnanam, J. (2023). A time series is worth 64 words: Long-term forecasting with Transformers. *Proceedings of the International Conference on Learning Representations (ICLR 2023)*.

[28] Zeng, A., Chen, M., Zhang, L., & Xu, Q. (2023). Are Transformers Effective for Time Series Forecasting? *Proceedings of the AAAI Conference on Artificial Intelligence*, *37*(9), 11121-11128. https://doi.org/10.1609/aaai.v37i9.26317

[29] Douaioui, K., Oucheikh, R., Benmoussa, O., & Mabrouki, C. (2024). Machine learning and deep learning models for demand forecasting in supply chain management: A critical review. *Applied System Innovation*, *7*(5), 93. https://doi.org/10.3390/asi7050093

[30] Xuan, W. & Deng, M. (2023). Logistics service quality sentiment analysis with deep attention LSTM. *Tehnicki vjesnik-Technical Gazette*, *30*(2), 634-641. https://doi.org/10.17559/TV-20221018031450

[31] Zhao, F., Dong, B., Sun, Y., & Huang, S. (2023). Forecast of railway freight volume using prophet-deep AR. *Tehnicki vjesnik-Technical Gazette*, *30*(4), 1126-1134. https://doi.org/10.17559/TV-20230226000383

Contact information:

Kyoungjong PARK
Department of Business Administration,
Gwangju University, South Korea
E-mail: kjpark@gwangju.ac.kr